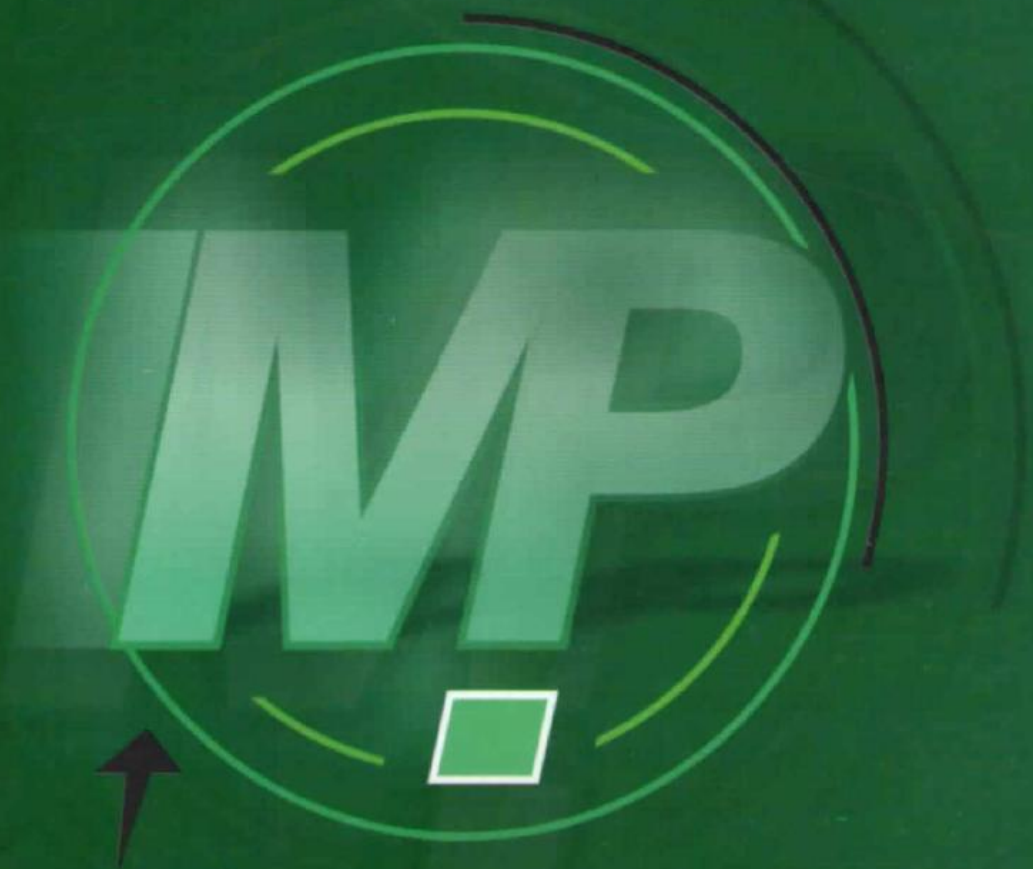
osvaldo cairó

Metodología de la programación

Algoritmos, diagramas de flujo y programas

3ª Edición



Alfaomega

www.FreeLibros.com

Metodología de la programación

Algoritmos, diagramas de flujo y programas

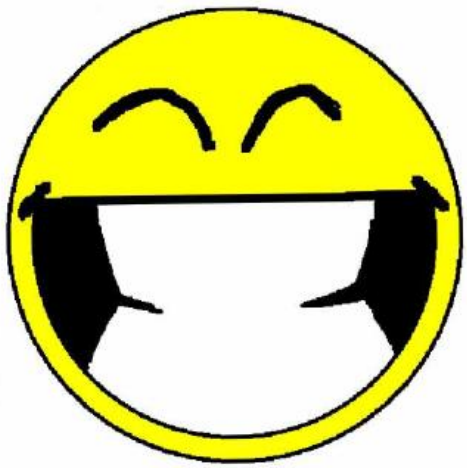
www.FreeLibros.com

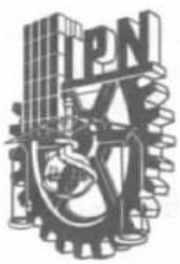
Metodología de la programación

Algoritmos, diagramas de flujo y programas

3a. edición

Dr. Osvaldo Cairó Battistutti
Profesor – Investigador del
Instituto Tecnológico Autónomo de México (ITAM)
Director del laboratorio KAMET
Miembro del Sistema Nacional de Investigadores (SNI), Nivel 1





Alfaomega

www.FreeLibros.com

© 2005 Osvaldo Cairó Battistutti

Segunda edición: México, julio 2003
Tercera edición: México, septiembre 2005
Primera reimpresión: México, abril 2006
Segunda reimpresión: México, agosto 2006

© 2005 ALFAOMEGA GRUPO EDITOR, S.A. de C.V.
Pitágoras 1139, Col. Del Valle, 03100 México, D.F.

Miembro de la Cámara Nacional de la Industria Editorial Mexicana
Registro No. 2317

Internet: **http://www.alfaomega.com.mx**
E-mail: **ventas1@alfaomega.com.mx**

ISBN 970-15-1100-X

Derechos reservados
Esta obra es propiedad intelectual de su autor y los derechos de publicación en lengua española han sido legalmente transferidos al editor. Prohibida su reproducción parcial o total por cualquier medio sin permiso por escrito del propietario de los derechos del copyright.

NOTA IMPORTANTE
La información contenida en esta obra tiene un fin exclusivamente didáctico y, por lo tanto, no está previsto su aprovechamiento a nivel profesional o industrial. Las indicaciones técnicas y programas incluidos, han sido elaborados con gran cuidado por el autor y reproducidos bajo estrictas normas de control. ALFAOMEGA GRUPO EDITOR, S.A. de C.V. no será jurídicamente responsable por: errores u omisiones; daños y perjuicios que se pudieran atribuir al uso de la información comprendida en este libro, ni por la utilización indebida que pudiera dársele.

Impreso en México - Printed in Mexico

www.FreeLibros.com

A mi hijo Facundo

www.FreeLibros.com

Colaboraron en la edición de esta obra:

Supervisión de edición

Martha Elena Figueroa Gutiérrez

Diagramación

Jesús García Álvarez

Producción:

Guillermo González Dorantes

www.FreeLibros.com

Prefacio a la tercera edición

Esta obra está dirigida a todos aquellos estudiantes que están iniciando sus estudios en carreras de computación, y que por tal razón necesitan resolver problemas cuya solución se plantea algorítmicamente. La literatura sobre temas introductorios en computación es en general escasa. Esta fue la razón y la motivación principal para desarrollar la tercera edición del libro *Metodología de la Programación*. Esta edición reúne lo mejor de las ediciones anteriormente publicadas con la misma editorial, e incluye nuevos problemas, mejores soluciones y novedosos componentes que permiten una mejor comprensión y aplicación de los conceptos estudiados. El lenguaje que se utiliza para mostrar la solución de los problemas es pseudo-código de alto nivel. Esta característica es muy importante porque facilita al alumno concentrarse en las estructuras de datos y algoritmos asociados a ellas sin la necesidad de relacionarlas a un lenguaje de programación en particular. Una vez que el alumno domina los conceptos, puede escribir estos programas muy fácilmente en un lenguaje de programación, como por ejemplo *C*.

Los problemas básicamente se descomponen en dos tipos, aquellos que requieren de una búsqueda y los que necesitan de un algoritmo para su solución. La inteligencia artificial se encarga de los problemas del primer tipo. Hay métodos que permiten resolver de manera eficiente, por ejemplo, puzzles y juegos. En este libro nos enfocaremos en los problemas del segundo tipo, aquellos que requieren de una solución algorítmica. Este tipo se caracteriza por requerir para su solución tanto de un análisis sistemático y profundo, como de un razonamiento estructurado y flexible. Indudablemente surgen algunos cuestionamientos al respecto:

¿Podemos enseñar a analizar un problema?
¿Podemos enseñar a razonar flexiblemente?
¿Podemos enseñar a pensar?

Las respuestas innegablemente no son fáciles de obtener. Sin embargo, el esfuerzo y la búsqueda constante del saber siempre valen la pena. Encontrar caminos es una tarea estimulante. Enseñar al alumno a ser flexible, a observar un problema desde ángulos diferentes, a ver distintas alternativas, a sentir la belleza

www.FreeLibros.com

de una solución es una tarea de maestros con pasión. *Enseñar a aprender. Enseñar a ver. Enseñar a saber.*

Respecto a la estructura de la obra, cabe señalar que ésta se divide en seis capítulos. En los cinco primeros se exponen las estructuras algorítmicas de control selectivas y repetitivas y las estructuras de datos, arreglos y registros. En el último capítulo se presenta un conjunto de problemas a resolver, agrupados según los temas tratados en los capítulos previos. El nivel de complejidad de cada uno de ellos aumenta en forma gradual. Los temas se exponen con amplitud y claridad. El aprendizaje se reafirma con una gran cantidad de ejercicios diseñados expresamente como elementos de ayuda para el análisis, el razonamiento, la práctica y el entendimiento de los conceptos. Se muestran también numerosas tablas en los primeros capítulos con seguimientos de los algoritmos para exponer cómo funcionan éstos y de qué manera afectan las estructuras de datos y de control que se estudian.

Muchos contribuyeron en forma diferente al desarrollo de este proyecto. Quiero agradecer especialmente al ITAM, a su gente, al Dr. Arturo Fernández Pérez, rector de esta gran institución, al Dr. Marcelo Mejía Olvera, Director de la División Académica de Ingenierías y al M.I. Rafael Gamboa Hirales, Jefe del Departamento Académico de Computación, por impulsar siempre proyectos de este tipo. Asimismo quiero agradecer a Lisandro Lazzaroni, Director de Staff en Santa Fe, Argentina, por su creatividad en el diseño de la portada.

Osvaldo Cairó
México, D. F.

www.FreeLibros.com

Contenido

Prefacio a la tercera edición ..... **vii**

1 Algoritmos, diagramas de flujo y programas **1**
1.1 Problemas y algoritmos 1
1.2 Diagramas de flujo 4
1.2.1 Reglas para la construcción de diagramas de flujo 4
1.3 Conceptos fundamentales 8
1.3.1 Tipos de datos 8
1.3.2 Identificadores, constantes y variables 9
1.3.3 Operaciones aritméticas 12
1.3.4 Expresiones lógicas 15
1.3.5 Bloque de asignación 22
1.4 Construcción de diagramas de flujo 24
1.5 Programas 31
Problemas resueltos 37

2 Estructuras algorítmicas selectivas **53**
2.1 Introducción 53
2.2 La estructura selectiva simple *si entonces* 54
2.3 La estructura selectiva doble *si entonces/sino* 57
2.4 La estructura selectiva múltiple *si múltiple* 62
2.5 Estructuras selectivas en cascada (anidadas) 70
Problemas resueltos 77

3 Estructuras algorítmicas repetitivas **107**
3.1 Introducción 107
3.2 La estructura repetitiva *repetir* (FOR) 107
3.3 La estructura repetitiva *mientras* (WHILE) 115
Problemas resueltos 121

4 Estructuras de datos: arreglos **175**
4.1 Introducción 175
4.2 Arreglos unidimensionales 181

www.FreeLibros.com

4.2.1 Definición de arreglos 182
4.2.2 Operaciones con arreglos 184
4.3 Arreglos multidimensionales 209
4.3.1 Arreglos bidimensionales 210
4.3.2 Arreglos de más de dos dimensiones 222
Problemas resueltos 229

5 Estructuras de datos: registros 343
5.1 Registros 343
5.1.1 Definición de registros 344
5.1.2 Acceso a los campos de un registro 346
5.1.3 Diferencias con arreglos 347
5.1.4 Combinaciones entre arreglos y registros 347
Problemas resueltos 354

6 Problemas suplementarios 401
Arreglos unidimensionales 441
Arreglos bidimensionales 446
Arreglos de más de dos dimensiones 453
Arreglos paralelos 454

Indice 463

www.FreeLibros.com

1

Algoritmos, diagramas de flujo y programas

1.1 Problemas y algoritmos

Casi inconscientemente, los humanos efectuamos cotidianamente una serie de pasos, procedimientos o acciones que nos permiten alcanzar un resultado o resolver un problema.

Esta serie de pasos, procedimientos o acciones, comenzamos a aplicarlas muy temprano en la mañana cuando, por ejemplo, decidimos tomar un baño. Posteriormente cuando pensamos en desayunar también seguimos una serie de pasos que nos permiten alcanzar un resultado específico: tomar el desayuno. La historia se repite innumerables veces durante el día. Continuamente seguimos una serie de pasos o conjunto de acciones que nos permiten alcanzar un resultado. Estamos en realidad aplicando un *algoritmo para resolver un problema.*

> *"Formalmente definimos un algoritmo como un conjunto de pasos, procedimientos o acciones que nos permiten alcanzar un resultado o resolver un problema".*

Muchas veces aplicamos el algoritmo de manera inadvertida, inconsciente o automáticamente. Esto generalmente se produce cuando el problema que tenemos enfrente lo hemos resuelto con anterioridad un gran número de veces.

Supongamos que simplemente tenemos que abrir una puerta. Lo hemos hecho tantas veces que difícilmente nos ponemos a enumerar los pasos para alcanzar este objetivo. Lo hacemos de manera automática. Lo mismo ocurre cuando queremos subirnos a un automóvil, cuando tenemos que lustrar nuestros zapa-

www.FreeLibros.com

tos, cuando nos calzamos, cuando nos vestimos, cuando tenemos desafortunadamente que cambiar la llanta de un automóvil o, simplemente cuando queremos tomar un vaso con agua.

Por otra parte, existe una gran cantidad de problemas que requieren de un análisis profundo y de un pensamiento flexible y estructurado para su solución. En este libro nos interesa abordar ese tipo de problemas. Invariablemente surgen ciertas preguntas:

- ¿Podemos enseñar a resolver un problema?
- ¿Podemos enseñar a analizar el mismo?
- ¿Podemos enseñar a pensar . . . . ?

Lógicamente las respuestas a estas interrogantes son difíciles de obtener. No existen reglas específicas que nos permitan resolver un problema. Sin embargo, creemos que se pueden ofrecer un conjunto de técnicas y herramientas metodológicas que permitan flexibilizar y estructurar el razonamiento utilizado en la solución de un problema. Eso provocará finalmente la construcción de algoritmos eficientes.

Ejemplo 1.1

Construya un algoritmo para preparar *"Pechugas de pollo en salsa de elote y chile poblano"*

Ingredientes (para 6 personas):
3 pechugas deshuesadas, sin piel y partidas a la mitad.
1 diente de ajo.
4 gramos de pimienta negra.
sal.
6 cucharadas de aceite.
5 chiles poblanos asados y limpios.
½ taza de leche.
¼ taza de crema ligera.
1 lata de crema de elote.

Algoritmo (Preparación):

- Muela el ajo, la pimienta y un poco de sal y únteselo a las pechugas.
- Caliente el aceite y dore las pechugas.
- Licue los chiles con la leche y la crema, y mézclelos con la crema de elote.
- En una fuente coloque las pechugas y báñelas con la mezcla anterior.
- Cubra el platón con papel aluminio y hornee a 200°C, durante 15 minutos.

Nota: El algoritmo fue probado repetidas veces y siempre arrojó el mismo resultado: *un platillo exquisito*.

www.FreeLibros.com

En la figura 1.1 podemos observar las etapas que debemos seguir para la solución de un problema.

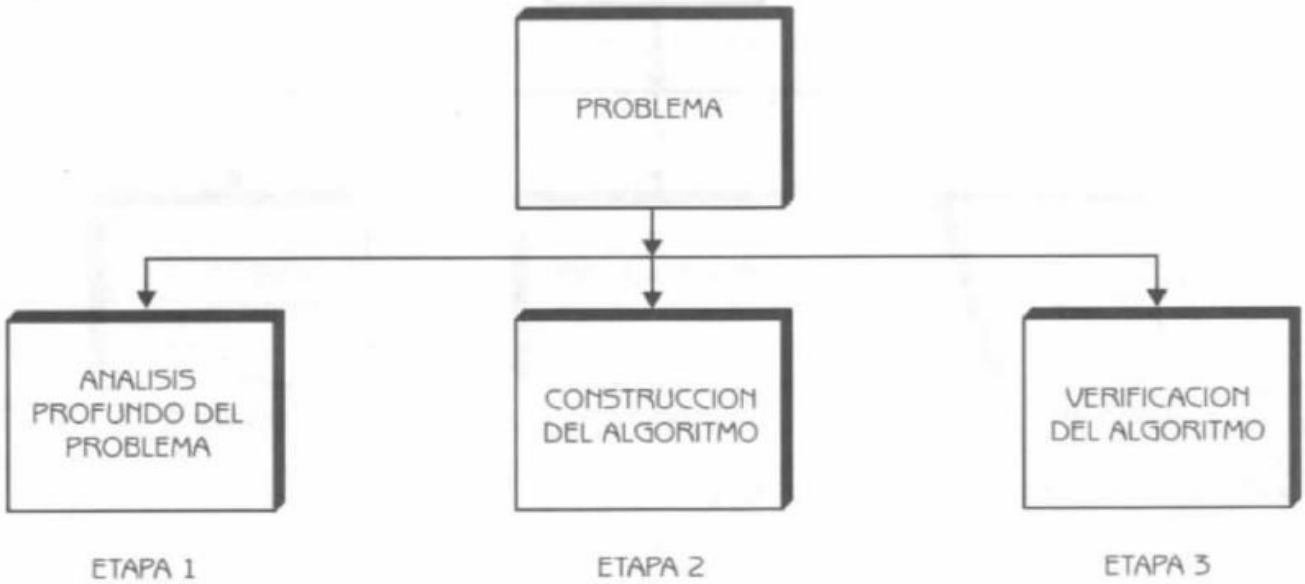


Figura 1.1 Etapas de la Solución de un Problema.

Nota: Por verificación del algoritmo entendemos el seguimiento del mismo con datos que sean representativos del problema que queremos resolver.

Las características que los algoritmos deben reunir son las siguientes:

Precisión: Los pasos a seguir en el algoritmo deben ser *precisados* claramente.

Determinismo: El algoritmo, dado un conjunto de datos idénticos de entrada, siempre debe arrojar los mismos resultados.

Finitud: El algoritmo, independientemente de la complejidad del mismo, siempre debe ser de longitud finita.

Por otra parte, un algoritmo consta de tres secciones o módulos principales. En la figura 1.2 podemos observar las secciones que constituyen un algoritmo.

El *módulo 1* representa la operación o acción que permite el ingreso de los datos del problema.

El *módulo 2* representa la operación o conjunto de operaciones secuenciales, cuyo objetivo es obtener la solución al problema.

El *módulo 3* representa una operación o conjunto de operaciones que permiten comunicar al exterior el o los resultados alcanzados.

www.FreeLibros.com

En la figura 1.1 podemos observar las etapas que debemos seguir para la solución de un problema.

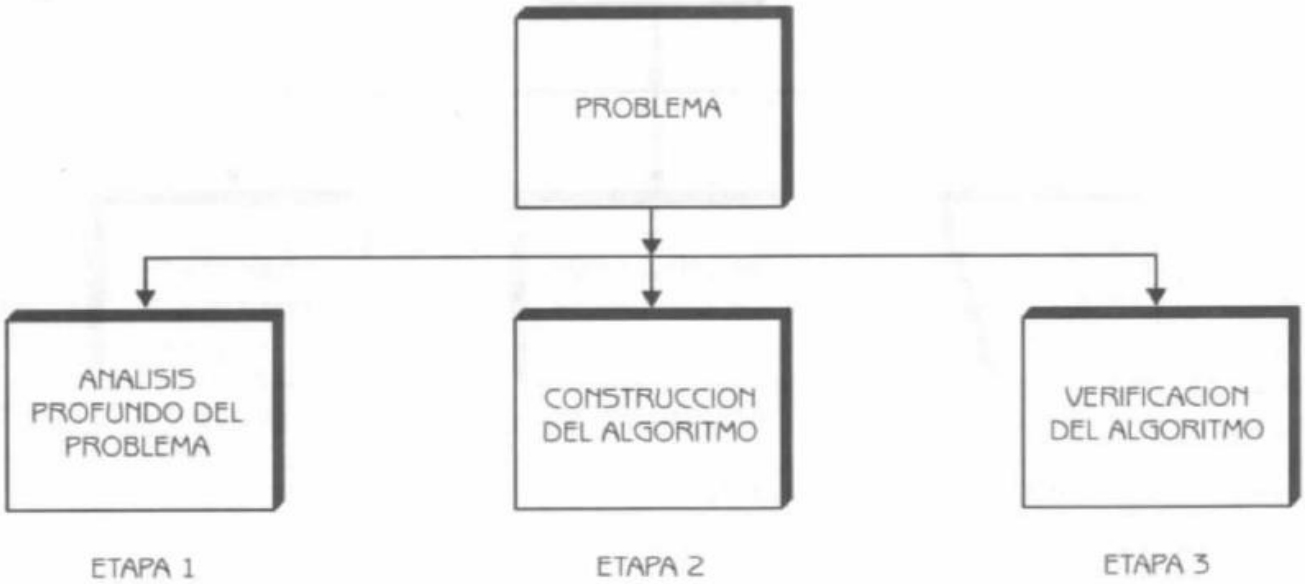


Figura 1.1 Etapas de la Solución de un Problema.

Nota: Por verificación del algoritmo entendemos el seguimiento del mismo con datos que sean representativos del problema que queremos resolver.

Las características que los algoritmos deben reunir son las siguientes:

Precisión: Los pasos a seguir en el algoritmo deben ser *precisados* claramente.

Determinismo: El algoritmo, dado un conjunto de datos idénticos de entrada, siempre debe arrojar los mismos resultados.

Finitud: El algoritmo, independientemente de la complejidad del mismo, siempre debe ser de longitud finita.

Por otra parte, un algoritmo consta de tres secciones o módulos principales. En la figura 1.2 podemos observar las secciones que constituyen un algoritmo.

El *módulo 1* representa la operación o acción que permite el ingreso de los datos del problema.

El *módulo 2* representa la operación o conjunto de operaciones secuenciales, cuyo objetivo es obtener la solución al problema.

El *módulo 3* representa una operación o conjunto de operaciones que permiten comunicar al exterior el o los resultados alcanzados.

www.FreeLibros.com

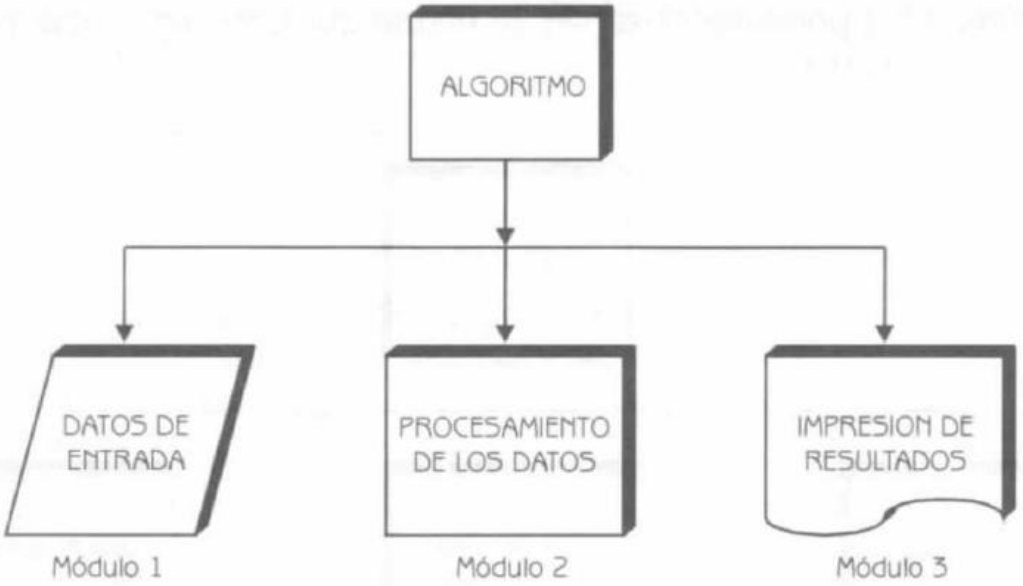


Figura 1.2 Módulos o Secciones de un Algoritmo.

1.2 Diagramas de flujo

Un diagrama de flujo representa la esquematización gráfica de un algoritmo. En realidad muestra gráficamente los pasos o procesos a seguir para alcanzar la solución de un problema. Su correcta construcción es sumamente importante porque a partir del mismo se escribe un programa en algún lenguaje de programación. Si el diagrama de flujo está completo y correcto, el paso del mismo a un lenguaje de programación es relativamente simple y directo.

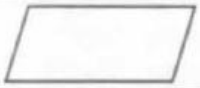
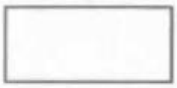
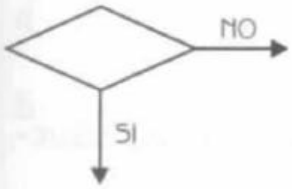
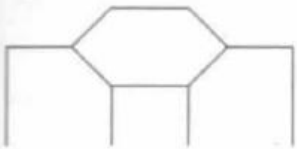
A continuación en la tabla 1.1 presentamos los símbolos que utilizaremos, y una explicación de los mismos. Estos satisfacen las recomendaciones de la "*International Organization for Standardization*" (ISO) y la "*American National Standards* Institute" (ANSI).

A continuación en la figura 1.3 presentamos las etapas que debemos seguir en la construcción de un diagrama de flujo.

1.2.1 Reglas para la construcción de diagramas de flujo

Debemos recordar que un diagrama de flujo debe ilustrar gráficamente los pasos o procesos a seguir para alcanzar la solución de un problema. Los símbolos presentados, colocados adecuadamente, permiten crear una estructura gráfica flexible que ilustra los pasos a seguir para alcanzar un resultado específico. El diagrama de flujo facilitará más tarde la escritura del programa en algún lenguaje de programación.

www.FreeLibros.com

Tabla 1.1 Símbolos utilizados en los Diagramas de Flujo	
Representación del Símbolo	**Explicación del Símbolo**
	Símbolo utilizado para marcar el *inicio* y el *fin* del diagrama de flujo.
	Símbolo utilizado para introducir los datos de entrada. Expresa *lectura*.
	Símbolo utilizado para representar un *proceso*. En su interior se expresan asignaciones, operaciones aritméticas, cambios de valor de celdas en memoria, etc.
	Símbolo utilizado para representar una decisión. En su interior se almacena una condición, y dependiendo del resultado de la evaluación de la misma se sigue por una de las ramas o caminos alternativos. Este símbolo se utiliza en la estructura selectiva *si entonces* que estudiaremos en el siguiente capítulo, y en las estructuras repetitivas *repetir* y *mientras* que analizaremos en el capítulo 3.
	Símbolo utilizado para representar la estructura selectiva doble *si entonces/sino*. En su interior se almacena una condición. Si el resultado es verdadero se continúa por el camino de la izquerda, y si es falso por el camino de la derecha.
	Símbolo utilizado para representar una decisión múltiple. En su interior se almacena un selector, y dependiendo del valor de dicho selector se sigue por una de las ramas o caminos alternativos. Este símbolo se utiliza en la estructura selectiva *si múltiple*, que analizaremos en el siguiente capítulo.
	Símbolo utilizado para representar la impresión de un resultado. Expresa *escritura*.
	Símbolos utilizados para expresar la dirección del flujo del diagrama.
	Símbolo utilizado para expresar conexión dentro de una misma página.
	Símbolo utilizado para expresar conexión entre páginas diferentes.
	Símbolo utilizado para expresar un módulo de un problema. En realidad expresa que para continuar con el flujo normal del diagrama debemos primero resolver el subproblema que enuncia en su interior.

www.FreeLibros.com

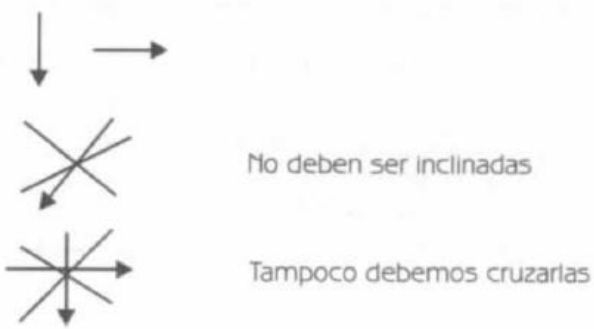
Nota:
Se debe observar que estas fases se presentan en la mayoría de los diagramas de flujo, aunque a veces en orden diferente o repitiendo alguna(s) de ellas. También es frecuente tener que realizar *toma de decisiones y repetir* una serie de pasos un número determinado o no de veces, pero estos conceptos serán analizados en capítulos posteriores.

Figura 1.3 Etapas en la Construcción de un Diagrama de Flujo.

A continuación presentamos un conjunto de reglas que permiten la construcción de diagramas de flujo.

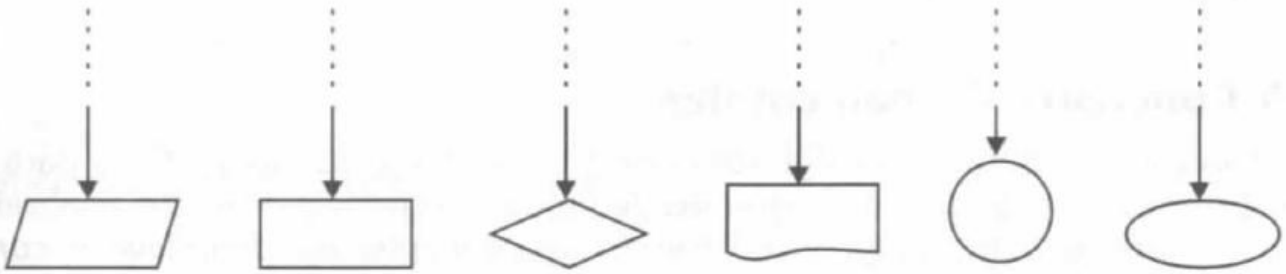
1. Todo diagrama de flujo debe tener un *inicio* y un *fin*.

2. Las líneas utilizadas para indicar la dirección del flujo del diagrama deben ser rectas, verticales y horizontales.

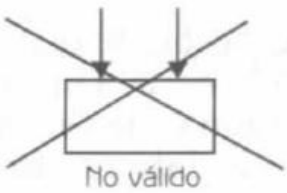


www.FreeLibros.com

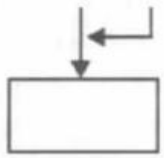
3. Todas las líneas utilizadas para indicar la dirección del flujo del diagrama deben estar conectadas. La conexión puede ser a un símbolo que exprese lectura, proceso, decisión, impresión, conexión o fin de diagrama.



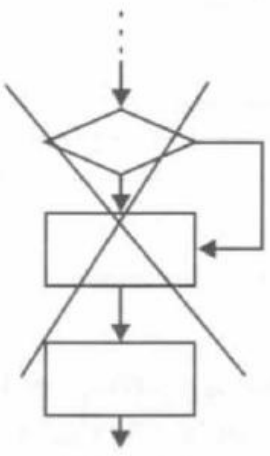
4. El diagrama de flujo debe ser construido de arriba hacia abajo (top–down) y de izquierda a derecha (right to left).
5. La notación utilizada en el diagrama de flujo debe ser independiente del lenguaje de programación. La solución presentada en el diagrama puede escribirse posteriormente y fácilmente en diferentes lenguajes de programación.
6. Es conveniente cuando realizamos una tarea compleja poner comentarios que expresen o ayuden a entender lo que hicimos.
7. Si el diagrama de flujo requiriera más de una hoja para su construcción, debemos utilizar los conectores adecuados y enumerar las páginas convenientemente.
8. No puede llegar más de una línea a un símbolo.



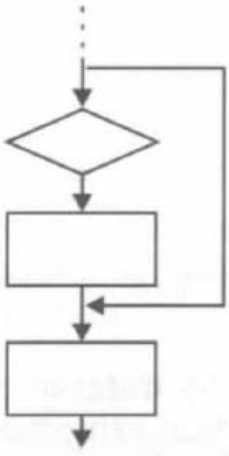
No válido



Válido



No válido



Válido

www.FreeLibros.com

A continuación nos gustaría mostrar algunos ejemplos con el objeto de que el lector se familiarice con la simbología presentada. Sin embargo, pensamos que antes es conveniente exponer algunos conceptos muy importantes que serán fundamentales para la construcción de los diagramas de flujo. Dejaremos los ejemplos para la sección 1.4.

1.3 Conceptos fundamentales

En esta sección trataremos algunos conceptos que son fundamentales para la construcción de algoritmos, diagramas de flujo y programas. Primero analizaremos los *tipos de datos*, luego estudiaremos los conceptos de *identificador, constantes y variables*, más adelante analizaremos las *operaciones aritméticas y expresiones lógicas*. Por último estudiaremos los *bloques de asignación*.

1.3.1 Tipos de datos

Los *datos* a procesar por una computadora pueden clasificarse en:

- Simples
- Estructurados

La principal característica de los datos simples es que ocupan sólo una casilla de memoria (Fig 1.4a), por lo tanto, una variable simple hace referencia a un único valor a la vez. Dentro de este grupo de datos se encuentran: enteros, reales, caracteres, booleanos, enumerados y subrangos (los dos últimos no existen en algunos lenguajes de programación).

Los datos estructurados se caracterizan por el hecho de que con un nombre (identificador de variable estructurada) se hace referencia a un grupo de casillas de memoria (Fig. 1.4b). Es decir, un dato estructurado tiene varios componentes. Cada uno de los componentes puede ser a su vez un dato simple o estructurado. Sin embargo, los componentes básicos (los del nivel más bajo) de cualquier tipo estructurado son datos simples. Dentro de este grupo de datos se encuentran: arreglos, cadena de caracteres, registros y conjuntos.

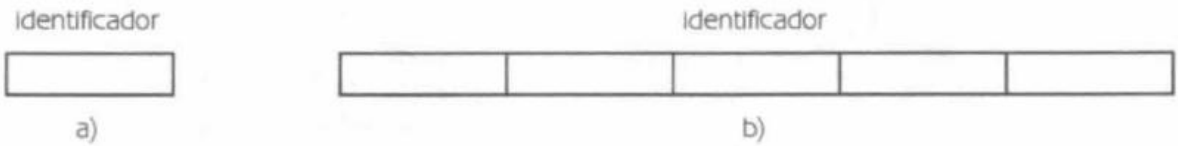


Figura 1.4 Datos simples y estructurados. a) Dato simple. b) Dato estructurado.

A continuación trataremos los datos simples: enteros, reales, caracteres y booleanos; y el dato estructurado: cadena de caracteres. Posteriormente en los capítulos 4 y 5, estudiaremos los datos estructurados arreglos y registros.

www.FreeLibros.com

Datos numéricos

Dentro de los tipos de datos numéricos encontramos los *enteros* y los *reales*. Los *enteros* son números que pueden estar precedidos del signo + o –, y que no tienen parte decimal. Por ejemplo:

128 1528 –714 8530 16235 –14780

Los *reales* son números que pueden estar precedidos del signo + o –, y que tienen una parte decimal. Por ejemplo:

7.5 128.0 –37.865 129.7 16000.50 –15.0

Datos alfanuméricos

Dentro de este tipo de datos encontramos los de tipo caracter (simple) y cadena de caracteres (estructurado). Son datos cuyo contenido pueden ser letras del abecedario (a,b,c,...,z), dígitos (0, 1, 2, ..., 9) o símbolos especiales (#, $, ^, *, %, /, !, +, –, ..., etc.). Debemos remarcar que aunque este tipo de datos pueden contener números, no pueden ser utilizados para realizar operaciones aritméticas.

Un dato tipo *caracter* contiene un solo caracter, y se escribe entre apóstrofes. Por ejemplo:

'a' 'B' '$' '9' '–' '#' 'f'

Un dato tipo *cadena de caracteres* contiene un conjunto de caracteres, y se escribe entre comillas. La longitud de una cadena depende de los lenguajes de programación, aunque normalmente se acepta una longitud máxima de 255.

"abcde" "$9#7" "Carlos Gómez" "Rosario" "754-27-22"

Datos lógicos

Dentro de este tipo de datos encontramos los *booleanos*. Son datos que sólo pueden tomar dos valores: verdadero (true) o falso (false).

1.3.2 Identificadores, constantes y variables

Identificadores

Los datos a procesar por una computadora, ya sean simples o estructurados, deben almacenarse en casillas o celdas de memoria para su posterior utilización. Estas casillas o celdas de memoria (constantes o variables) tienen un *nombre* que permite su identificación.

Llamaremos dentificador al nombre que se les da a las casillas de memoria. Un identificador se forma de acuerdo a ciertas reglas (las mismas pueden tener alguna variante dependiendo del lenguaje de programación utilizado):

www.FreeLibros.com

- El primer caracter que forma un identificador debe ser una letra (a, b, c, ..., z).
- Los demás caracteres pueden ser letras (a,b,c,...,z), dígitos (0,1,2,...,9) o el siguiente símbolo especial: _ .
- La longitud del identificador es igual a 7 en la mayoría de los lenguajes de programación.

A continuación, en la figura 1.5, podemos observar ejemplos de identificadores.

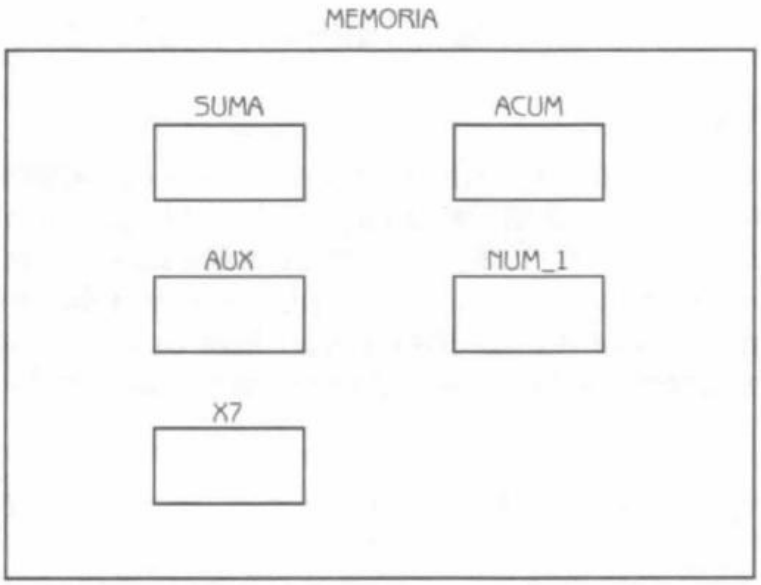


Figura 1.5 Casillas de Memoria con los Nombres de Identificadores.

Constantes

Las *constantes* son datos que no cambian durante la ejecución de un programa. Para nombrar las constantes utilizamos los identificadores que mencionamos anteriormente. Existen tipos de constantes como tipos de datos, por lo tanto, puede haber constantes de tipo entero, real, caracter, cadena de caracteres, etc.

Observe que en la figura 1.6, la constante NUM es de tipo entero, NREAL y NUMREA son de tipo real, y RESU de tipo cadena de caracteres. Estas constantes no cambiarán su valor durante la ejecución del programa. Es muy importante que los nombres de las constantes sean representativas de la función que tienen las mismas en el programa.

Variables

Las variables son objetos que pueden cambiar su valor durante la ejecución de un programa. Para nombrar las variables utilizaremos los identificadores que hemos explicado con anterioridad. Al igual que las constantes, pueden existir tipos de variables como tipos de datos.

www.FreeLibros.com

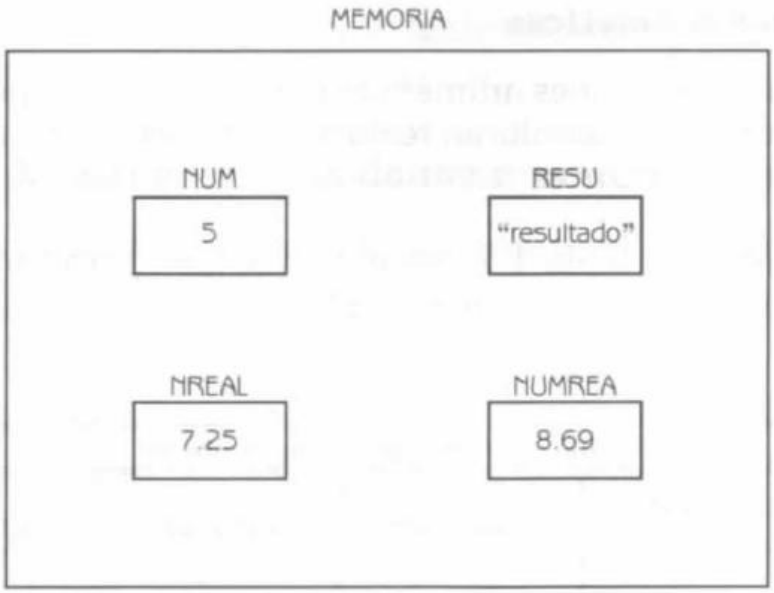


Figura 1.6 Constantes representadas en la Memoria.

En la figura 1.7, la variable I es de tipo entero, tiene un valor inicial de cero y cambiará su valor durante la ejecución del programa. Las variables SUEL y SUMA son de tipo real, están inicializadas con el valor de cero, y al igual que la variable I, seguramente cambiarán su valor durante la ejecución del programa.

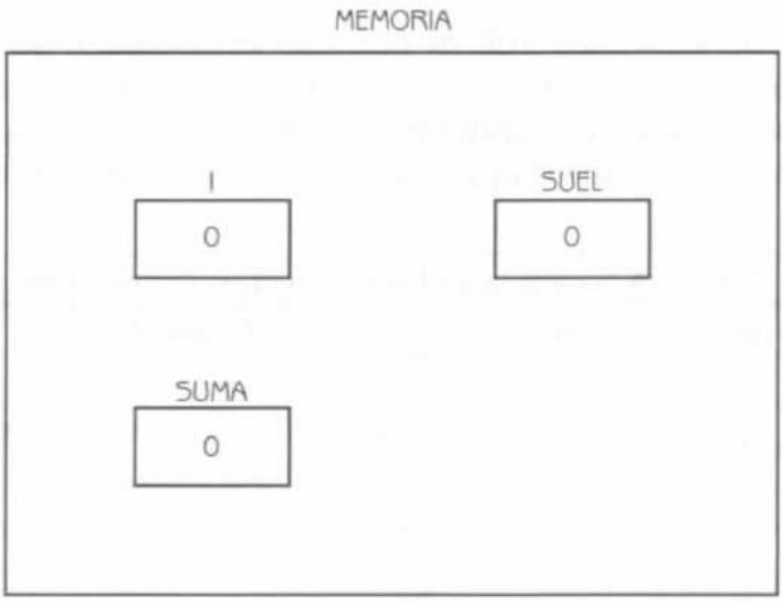


Figura 1.7 Variables representadas en la Memoria.

Debemos remarcar que los nombres de las variables deben ser representativos de la función que cumplen en el programa.

www.FreeLibros.com

1.3.3 Operaciones aritméticas

Para poder realizar operaciones aritméticas necesitamos de operadores aritméticos. Estos operadores nos permitirán realizar operaciones aritméticas entre operandos: números, constantes o variables. El resultado de una operación aritmética será un número.

A continuación en la tabla 1.2 presentamos los operadores aritméticos, la operación que pueden realizar, un ejemplo de su uso y el resultado de dicho ejemplo.

Tabla 1.2 Operadores Aritméticos

Operador Aritmético	Operación	Ejemplo	Resultado
**	Potencia	4**3	64
*	Multiplicación	8.25*7	57.75
/	División	15/4	3.75
+	Suma	125.78 + 62.50	188.28
–	Resta	65.30 – 32.33	32.97
mod	Módulo (residuo)	15 mod 2	1
div	División entera	17 div 3	5

Al evaluar expresiones que contienen operadores aritméticos debemos respetar la jerarquía en el orden de aplicación. Es decir, si tenemos en una expresión más de un operador, debemos aplicar primero el operador de mayor jerarquía, resolver esa operación, y así sucesivamente. Es importante señalar que el operador () es un operador asociativo que tiene la prioridad más alta en cualquier lenguaje de programación. En la tabla 1.3 se presenta la jerarquía de los operadores.

Tabla 1.3 Jerarquía de los Operadores Aritméticos

Operador	Jerarquía	Operación
**	(mayor)	Potencia
*,/,mod,div	↓	Multiplicación, división, módulo, división entera
+,–	(menor)	Suma, resta

Las reglas para resolver una expresión aritmética son las siguientes:

1. Si una expresión contiene subexpresiones entre paréntesis, éstas se evalúan primero; respetando claro está la jerarquía de los operadores aritméticos en esta subexpresión. Si las subexpresiones se encuentran anidadas por paréntesis, primero se evalúan las subexpresiones que se encuentran en el último nivel de anidamiento.

www.FreeLibros.com

2. Los operadores aritméticos se aplican teniendo en cuenta la jerarquía y de izquierda a derecha.

Ejemplo 1.2

A continuación en este ejemplo presentamos varios casos y la forma de resolver los mismos.

Caso a)

$$\underbrace{7 + 5}_{1} - 6$$

$$\underbrace{12 - 6}_{2}$$

$$6$$

Caso b)

$$9 + \underbrace{7 * 8}_{1} - 36 / 5$$

$$9 + 56 - \underbrace{36 / 5}_{2}$$

$$\underbrace{9 + 56}_{3} - 7.2$$

$$\underbrace{65 - 7.2}_{4}$$

$$57.8$$

Caso c)

$$7 * \underbrace{5 ** 3}_{1} / 4 \text{ div } 3$$

$$\underbrace{7 * 125}_{2} / 4 \text{ div } 3$$

www.FreeLibros.com

$$\underbrace{875 / 4 \text{ div } 3}_{3}$$

$$\underbrace{218.75 \text{ div } 3}_{4}$$

$$72$$

Caso d)

$$7 * 8 * (160 \bmod \underbrace{3 ** 3}_{1}) \text{ div } 5 * 13 - 28$$

$$7 * 8 * (\underbrace{160 \bmod 27}_{2}) \text{ div } 5 * 13 - 28$$

$$\underbrace{7 * 8}_{3} * 25 \text{ div } 5 * 13 - 28$$

$$\underbrace{56 * 25}_{4} \text{ div } 5 * 13 - 28$$

$$\underbrace{1400 \text{ div } 5}_{5} * 13 - 28$$

$$\underbrace{280 * 13}_{6} - 28$$

$$\underbrace{3640 - 28}_{7}$$

$$3612$$

Caso e)

$$15 / 2 * (7 + (68 - 15 * 33 + (\underbrace{45 ** 2}_{1} / 16) / 3) / 15) + 19$$

$$15 / 2 * (7 + (68 - 15 * 33 + (\underbrace{2025 / 16}_{2}) / 3) / 15) + 19$$

www.FreeLibros.com

$$15 / 2 * (7 + (68 - \underbrace{15 * 33}_{3} + 126.5625 / 3) / 15) + 19$$

$$15 / 2 * (7 + (68 - 495 + \underbrace{126.5625 / 3}_{4}) / 15) + 19$$

$$15 / 2 * (7 + (\underbrace{68 - 495}_{5} + 42.1875) / 15) + 19$$

$$15 / 2 * (7 + (\underbrace{- 427 + 42.1875}_{6}) / 15) + 19$$

$$15 / 2 * (7 + \underbrace{(-) 384.8125 / 15}_{7}) + 19$$

$$15 / 2 * (\underbrace{7 + (-) 25.6541}_{8}) + 19$$

$$\underbrace{15 / 2}_{9} * (-) 18.6541 + 19$$

$$\underbrace{7.5 * (-) 18.6541}_{10} + 19$$

$$\underbrace{- 139.9062 + 19}_{11}$$

$$-120.9062$$

1.3.4 Expresiones lógicas

Las *expresiones lógicas o booleanas*, llamadas así en honor del matemático George Boole, están constituidas por números, constantes o variables y operadores lógicos o relacionales. El valor que pueden tomar estas expresiones es el de *verdadero* o *falso*. Se utilizan frecuentemente en las estructuras selectivas (dependiendo del resultado de la evaluación se toma por un determinado camino alternativo) y en las estructuras repetitivas (dependiendo del resultado de la evaluación se continúa con el ciclo o se interrumpe al mismo).

www.FreeLibros.com

Operadores relacionales

Los operadores relacionales son operadores que permiten comparar dos operandos. Los operandos pueden ser números, alfanuméricos, constantes o variables. Las constantes o variables, a su vez, pueden ser de tipo entero, real, caracter o cadena de caracteres. El resultado de una expresión con operadores relacionales es verdadero o falso.

A continuación en la tabla 1.4 presentamos los operadores relacionales, la operación que pueden realizar, un ejemplo de su uso y el resultado de dicho ejemplo.

Tabla 1.4 Operadores Relacionales

Operador	Operación	Ejemplo	Resultado
=	Igual que	'hola' = 'lola'	FALSO
< >	Diferente a	'a' <> 'b'	VERDADERO
<	Menor que	7 < 15	VERDADERO
>	Mayor que	22 > 11	VERDADERO
<=	Menor o igual que	15 <= 22	VERDADERO
>=	Mayor o igual que	35 >= 20	VERDADERO

Ejemplo 1.3

En este ejemplo presentamos varios casos de expresiones lógicas con operadores relacionales y la forma de resolver las mismas.

Caso a)

```
A = 5
B = 16

(A ** 2) > (B * 2)
 1

25 > (B * 2)
      2

25 > 32
 3

FALSO
```

www.FreeLibros.com

Caso b)

$$X = 6$$
$$B = 7.8$$

$$(\ X\ *\ 5\ +\ \underbrace{B\ **\ 3}_{1}\ /\ 4\ )\ <=\ (\ X\ **\ 3\ \text{div}\ B\ )$$

$$(\ \underbrace{X\ *\ 5}_{2}\ +\ 474.552\ /\ 4\ )\ <=\ (\ X\ **\ 3\ \text{div}\ B\ )$$

$$(\ 30\ +\ \underbrace{474.552\ /\ 4}_{3}\ )\ <=\ (\ X\ **\ 3\ \text{div}\ B\ )$$

$$(\ \underbrace{30\ +\ 118.638}_{4}\ )\ <=\ (\ X\ **\ 3\ \text{div}\ B\ )$$

$$148.638\ <=\ (\ \underbrace{X\ **\ 3}_{5}\ \text{div}\ B\ )$$

$$148.638\ <=\ (\ \underbrace{216\ \text{div}\ B}_{6}\ )$$

$$\underbrace{148.638\ <=\ 27}_{7}$$

FALSO

Caso c)

$$((1580\ \text{mod}\ 6\ *\ \underbrace{2\ **\ 7}_{1}\ )\ >\ (7\ +\ 8\ *\ 3\ **\ 4))\ >\ ((15\ *\ 2)\ =\ (60\ *\ 2\ /\ 4))$$

$$((\underbrace{1580\ \text{mod}\ 6}_{2}\ *\ 128\ )\ >\ (7\ +\ 8\ *\ 3\ **\ 4))\ >\ ((15\ *\ 2)\ =\ (60\ *\ 2\ /\ 4))$$

$$((\underbrace{2\ *\ 128}_{3}\ )\ >\ (7\ +\ 8\ *\ 3\ **\ 4))\ >\ ((15\ *\ 2)\ =\ (60\ *\ 2\ /\ 4))$$

www.FreeLibros.com

$$(256 > (7 + \underbrace{8 \cdot 3 \cdot\cdot 4}_{4})) > ((15 \cdot 2) = (60 \cdot 2 / 4))$$

$$(256 > (\underbrace{7 + 8 \cdot 81}_{5})) > ((15 \cdot 2) = (60 \cdot 2 / 4))$$

$$(256 > \underbrace{(7 + 648)}_{6}) > ((15 \cdot 2) = (60 \cdot 2 / 4))$$

$$\underbrace{(256 > 655)}_{7} > ((15 \cdot 2) = (60 \cdot 2 / 4))$$

$$FALSO > (\underbrace{(15 \cdot 2)}_{8} = (60 \cdot 2 / 4))$$

$$FALSO > (30 = \underbrace{(60 \cdot 2 / 4)}_{9})$$

$$FALSO > (30 = \underbrace{(120 / 4)}_{10})$$

$$FALSO > \underbrace{(30 = 30)}_{11}$$

$$FALSO > VERDADERO$$

$$FALSO$$

Nota: Cuando se utilizan los operadores de relación con operandos lógicos, *falso* es menor que *verdadero*.

Operadores lógicos

Los *operadores lógicos* son operadores que permiten formular condiciones complejas a partir de condiciones simples. Los operadores lógicos son de conjunción (y), disyunción (o) y negación (no). En la tabla 1.5 presentamos el operador lógico, la expresión lógica y significado de dicha expresión, teniendo en cuenta la jerarquía correspondiente.

www.FreeLibros.com

Tabla 1.5 Operadores Lógicos			
Operador lógico	**Jerarquía**	**Expresión lógica**	**Significado**
NO	(mayor)	No P	NO P No es cierto que P Es FALSO que P
Y	↓	P y Q	P ^ Q P sin embargo Q
O	(menor)	P o Q	P o Q o P o Q o ambas Mínimo P o Q

A continuación en la tabla 1.6 presentamos la tabla de verdad de los operadores lógicos.

Tabla 1.6 Tabla de verdad de los Operadores Lógicos					
P	**Q**	**~P**	**~Q**	**P o Q**	**P ^ Q**
VERDADERO	VERDADERO	FALSO	FALSO	VERDADERO	VERDADERO
VERDADERO	FALSO	FALSO	VERDADERO	VERDADERO	FALSO
FALSO	VERDADERO	VERDADERO	FALSO	VERDADERO	FALSO
FALSO	FALSO	VERDADERO	VERDADERO	FALSO	FALSO

Por último en la tabla 1.7 presentamos la jerarquía correspondiente de todos los operadores (aritméticos, relacionales y lógicos).

Tabla 1.7 Jerarquía de los Operadores	
Operadores	**Jerarquía**
( )	(mayor)
**	↓
*, /, div, mod	
=, < >, <, >, <=, >=	
NO	
Y	
O	(menor)

Nota: El operador () es un operador asociativo que tiene la prioridad más alta en cualquier lenguaje de programación. Por otra parte, debemos señalar que en ciertos lenguajes, las prioridades de los operadores se manejan en forma diferente. Por ejemplo, el operador lógico de negación *NO* en el lenguaje de programación C tiene la prioridad más alta después del operador asociativo.

www.FreeLibros.com

Ejemplo 1.4

A continuación en este ejemplo presentamos varios casos y la forma de resolver los mismos.

Caso a)

NO (15 >= 7 ** 2) O (43 – 8 * 2 div 4 <> 3 * 2 div 2)
1

NO (15 >= 49) O (43 – 8 * 2 div 4 <> 3 * 2 div 2)
2 3

NO FALSO O (43 – 16 div 4 <> 3 * 2 div 2)
4

NO FALSO O (43 – 4 <> 3 * 2 div 2)
5

NO FALSO O (43 – 4 <> 6 div 2)
6

NO FALSO O (43 – 4 <> 3)
7

NO FALSO O (39 <> 3)
8

NO FALSO O VERDADERO
9

VERDADERO O VERDADERO
10

VERDADERO

Caso b)

(15 >= 7 * 3 ** 2 Y 8 > 3 Y 15 > 6) O NO (7 * 3 < 5 + 12 * 2 div 3 ** 2)
1

www.FreeLibros.com

$$(15 >= \underbrace{7 * 9}_{2} \text{ Y } 8 > 3 \text{ Y } 15 > 6) \text{ O NO } (7 * 3 < 5 + 12 * 2 \text{ div } 3 ** 2)$$

$$(\underbrace{15 >= 63}_{3} \text{ Y } \underbrace{8 > 3}_{4} \text{ Y } \underbrace{15 > 6}_{5}) \text{ O NO } (7 * 3 < 5 + 12 * 2 \text{ div } 3 ** 2)$$

$$(\underbrace{\text{FALSO Y VERDADERO}}_{6} \text{ Y VERDADERO}) \text{ O NO } (7 * 3 < 5 + 12 * 2 \text{ div } 3 ** 2)$$

$$(\underbrace{\text{FALSO Y VERDADERO}}_{7}) \text{ O NO } (7 * 3 < 5 + 12 * 2 \text{ div } 3 ** 2)$$

$$\text{FALSO O NO } (\underbrace{7 * 3}_{9} < 5 + \underbrace{12 * 2}_{10} \text{ div } \underbrace{3 ** 2}_{8})$$

$$\text{FALSO O NO } (21 < 5 + \underbrace{24 \text{ div } 9}_{11})$$

$$\text{FALSO O NO } (21 < \underbrace{5 + 2}_{12})$$

$$\text{FALSO O NO } (\underbrace{21 < 7}_{13})$$

$$\text{FALSO O } \underbrace{\text{NO FALSO}}_{14}$$

$$\underbrace{\text{FALSO O VERDADERO}}_{15}$$

VERDADERO

Caso c)

$$\text{NO } ((\underbrace{7 * 3}_{1} \text{ div } 2 * 4) > (15 / 2 * 6 >= 15 * 2 / 17 = 15))$$

$$\text{NO } ((\underbrace{21 \text{ div } 2}_{2} * 4) > (15 / 2 * 6 >= 15 * 2 / 17 = 15))$$

www.FreeLibros.com

NO ((10 * 4) > (15 / 2 * 6 >= 15 * 2 / 17 = 15))
3

NO (40 > (15 / 2 * 6 >= 15 * 2 / 17 = 15))
4

NO (40 > (7.5 * 6 >= 15 * 2 / 17 = 15))
5 6

NO (40 > (45 >= 30 / 17 = 15))
7

NO (40 > (45 >= 1.75 = 15))
8

NO (40 > (VERDADERO = 15))
9

Error

Nota: No se puede realizar la comparación entre un valor lógico y un numérico, utilizando un operador relacional.

1.3.5 Bloque de asignación

Un bloque de asignación se utiliza para asignar valores o expresiones a una variable. La asignación es una operación destructiva. Esto significa que si la variable tenía asignado un valor, éste se destruye, conservando ahora el nuevo valor. El formato de la asignación es el siguiente:

Variable ← expresión o valor

Donde: expresión puede ser aritmética o lógica, o una constante o variable.

Observemos a continuación el siguiente ejemplo:

www.FreeLibros.com

Ejemplo 1.5

Supongamos que las variables I, ACUM y J son de tipo entero, REA y SUM de tipo real, CAR de tipo caracter y BAND de tipo booleano. Consideremos también que tenemos que realizar las siguientes asignaciones:

1. I ⟵ 0
2. I ⟵ I + 1
3. ACUM ⟵ 0
4. J ⟵ 5 ** 2 div 3
5. CAR ⟵ 'a'
6. ACUM ⟵ J div I
7. REA ⟵ ACUM / 3
8. BAND ⟵ (8 > 5) y (15 < 2 ** 3)
9. SUM ⟵ ACUM * 5 / J ** 2
10. I ⟵ I * 3
11. REA ⟵ REA / 5
12. BAND ⟵ BAND o (I = J)
13. I ⟵ REA
14. CAR ⟵ J

En la tabla 1.8 podemos observar los valores que van tomando las variables en memoria.

Tabla 1.8 Memoria

Número de Asignación	I	J	ACUM	REA	SUM	CAR	BAND
1	~~0~~						
2	~~1~~						
3			~~8~~				
4		8					
5						~~a~~	
6			8				
7				~~2.66~~			
8							~~FALSO~~
9					0.625		
10	~~3~~						
11				0.532			
12							FALSO
13	Error						
14						Error	

www.FreeLibros.com

Nota: Observe el lector que en la asignación número 13 a la variable I se le asigna una variable de tipo real, por lo que se produce un error. Lo mismo ocurre en la asignación número 14, a la variable tipo caracter *CAR* se le asigna una variable de tipo entero, por lo que también se produce un error.

1.4 Construcción de diagramas de flujo

Hasta el momento el lector ha estudiado algunos conceptos que le permiten construir algunos diagramas de flujo. Reforzaremos estos conceptos con ejemplos seleccionados. Es nuestro interés que el lector comience a desarrollar habilidad y una capacidad de razonamiento estructurada y flexible que le permita, en la medida que practique, obtener la solución a los problemas planteados.

Ejemplo 1.6

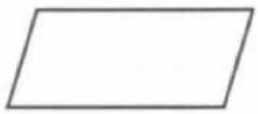
Construya un diagrama de flujo tal que dado los datos A, B, C y D que representan números enteros, escriba los mismos en orden inverso.

Datos: A, B, C, D (variables de tipo entero).

Consideraciones:

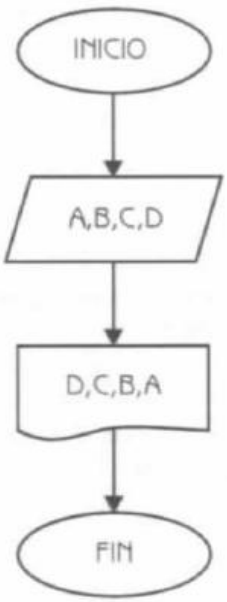
- Para el *inicio* y *fin* del diagrama de flujo se utiliza el símbolo:

- Para *lectura* se utiliza el símbolo:



- Para *escritura* se utiliza el símbolo:

www.FreeLibros.com



{Se leen los datos}

{Se escriben los datos en orden inverso}

Diagrama de Flujo 1.1

Observe el lector que si se ingresan los datos: *7*, *28*, *150* y *35*, la impresión produce lo siguiente: *35*, *150*, *28*, *7*.

Ejemplo 1.7

Construya un diagrama de flujo tal que dado los datos enteros A y B, escriba el resultado de la siguiente expresión:

$$\frac{(A + B)^2}{3}$$

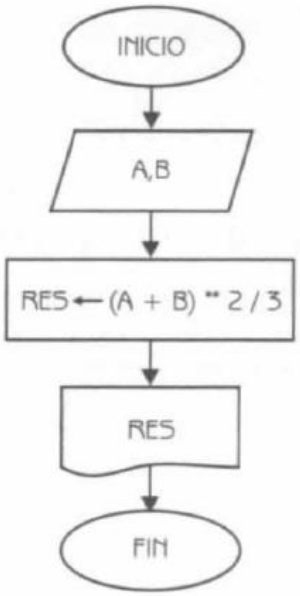
Datos: A, B (variables de tipo entero).

Recuerde lo siguiente:

- Para realizar un *proceso* se utiliza el símbolo:
- Para asignar una expresión o valor a una variable, se utiliza un *bloque de asignación*:

Variable ← expresión o valor

www.FreeLibros.com



{Lectura de datos}

{El resultado de la expresión se almacena en la variable de tipo real *RES*}

{Escritura de la variable *RES* que almacena el resultado de la expresión}

Diagrama de Flujo 1.2

Explicación de las variables

A, *B*: Variables de tipo entero.

RES: Variable de tipo real. Almacena el resultado de la expresión.

En la tabla 1.9 el lector podrá observar los datos que se ingresan y el resultado obtenido, para 5 corridas diferentes.

Tabla 1.9

NUMERO DE CORRIDA	DATOS		RESULTADO
	A	B	RES
1	5	6	40.33
2	7	10	96.33
3	0	3	3.00
4	12	2	65.33
5	14	–5	27.00

: Expresa valores que se imprimen

www.FreeLibros.com

Ejemplo 1.8

Dada la matrícula y 5 calificaciones de un alumno obtenidas a lo largo del semestre, construya un diagrama de flujo que imprima la matrícula del alumno y el promedio de sus calificaciones.

Datos: MAT, CAL1, CAL2, CAL3, CAL4, CAL5

Donde:

MAT	es una variable de tipo entero que representa la matrícula del alumno.
CAL1, CAL2, CAL3, CAL4 y CAL5	son variables de tipo real que representan las 5 calificaciones del alumno.

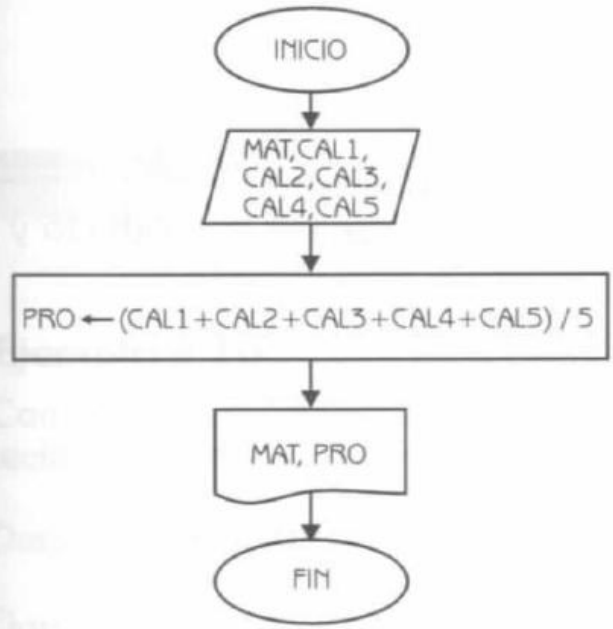


Diagrama de flujo 1.3

Explicación de las variables

MAT:	Variable de tipo entero.
CAL1, CAL2, CAL3, CAL4, CAL5:	Variables de tipo real.
PRO:	Variable de tipo real. Almacena el promedio de las calificaciones del alumno.

www.FreeLibros.com

En la tabla 1.10 el lector podrá observar los datos y resultados para 5 corridas diferentes.

Tabla 1.10

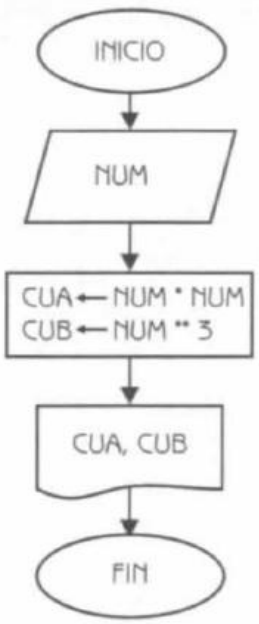
NUMERO DE CORRIDA	DATOS						RESULTADOS	
	MAT	CAL1	CAL2	CAL3	CAL4	CAL5	MAT	PRO
1	16500	8	8.5	9	7	6	16500	7.7
2	16650	9	8	9	7	9	16650	8.4
3	17225	9	10	10	8	9	17225	9.2
4	17240	8.5	9	7.5	6	6.5	17240	7.5
5	18240	7.3	6.8	9.5	8	8.5	18240	8.02

: Expresa valores que se imprimen

Ejemplo 1.9

Escriba un diagrama de flujo que permita calcular e imprimir el cuadrado y el cubo de un número entero positivo NUM.

Dato: NUM (variable de tipo entero).



{Lectura del dato}

{Cálculo del cuadrado y cubo del número *NUM* que se almacena en *CUA* y *CUB*, respectivamente}

{Impresión de los resultados}

Diagrama de flujo 1.4

www.FreeLibros.com

Explicación de las variables

NUM: Variable de tipo entero.
CUA: Variable de tipo real. Almacena el cuadrado del número que se ingresa.
CUB: Variable de tipo real. Almacena el cubo del número que se ingresa.

En la tabla 1.11 podemos observar el seguimiento del algoritmo para diferentes corridas.

Tabla 1.11

NUMERO DE CORRIDA	DATO	RESULTADOS	
	NUM	CUA	CUB
1	7	49	343
2	15	225	3375
3	8	64	512
4	12	144	1728
5	30	900	27000

: Expresa valores que se imprimen

Ejemplo 1.10

Construya un diagrama de flujo tal que dado como datos la base y la altura de un rectángulo, calcule el perímetro y la superficie del mismo.

Datos: BASE, ALTU

Donde:

BASE es una variable de tipo real que representa la base de un rectángulo.
ALTU es una variable de tipo real que indica la altura del rectángulo.

Recuerde que:

- La superficie de un rectángulo se calcula aplicando la siguiente fórmula:

Superficie = base * altura

Fórmula 1.1

www.FreeLibros.com

- El perímetro se calcula como:

$$\text{Perímetro} = 2 * (\text{base} + \text{altura})$$

Fórmula 1.2

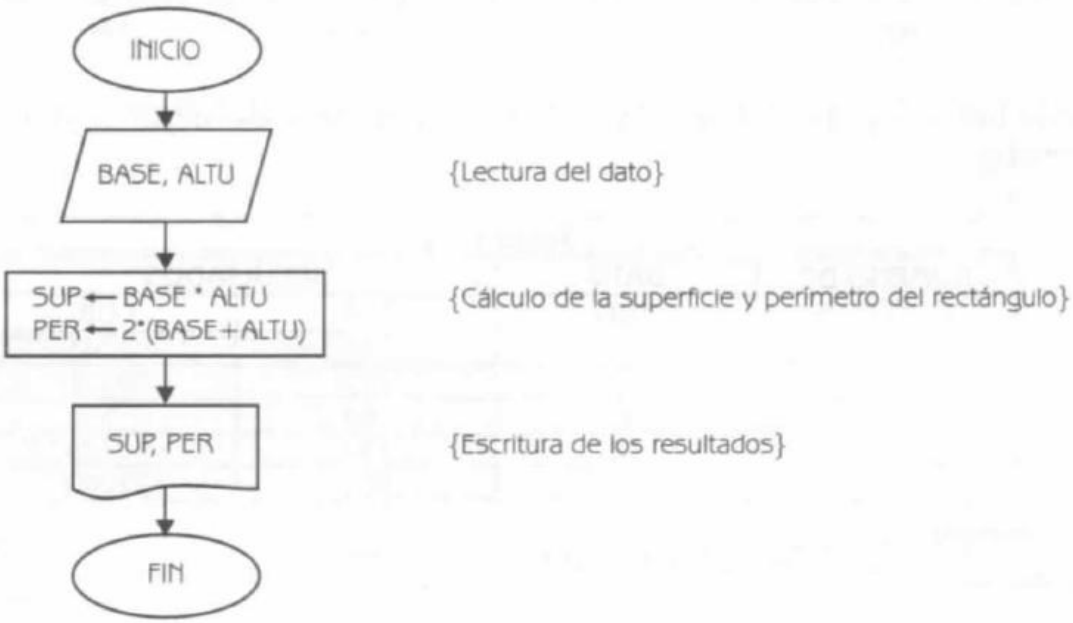


Diagrama de Flujo 1.5

Explicación de las variables

BASE, ALTU: Variables de tipo real.

SUP: Variable de tipo real. Almacena la superficie del rectángulo.

PER: Variable de tipo real. Almacena el perímetro del rectángulo.

Tabla 1.12

NUMERO DE CORRIDA	DATOS		RESULTADOS	
	BASE	ALTU	SUP	PER
1	8.5	6.2	52.70	29.40
2	7.9	15.3	120.87	46.40
3	15.18	22.0	333.96	74.36
4	12.63	7.9	99.77	41.06
5	39.40	68.5	2698.90	215.80

: Expresa valores que se imprimen

www.FreeLibros.com

1.5 Programas

Un *programa*, concepto desarrollado por Von Neumann en 1946, es un conjunto de instrucciones que sigue la computadora para alcanzar un resultado específico. El programa se escribe en un lenguaje de programación a partir de un diagrama de flujo diseñado con anterioridad.

Un *lenguaje de programación*, por otra parte, está constituido por un conjunto de reglas sintácticas (especifica la formación de instrucciones válidas) y semánticas (especifica el significado de estas instrucciones), que hacen posible escribir un programa.

Cientos de lenguajes de programación se han desarrollado hasta la fecha. Ya en 1969, se habían enumerado más de 120 que habían sido utilizados ampliamente. Actualmente a los lenguajes de programación podemos clasificarlos teniendo en cuenta el *tipo de problemas* que son capaces de resolver de manera natural. Así tenemos, por ejemplo, para programación estructurada: PASCAL, C, BASIC, FORTRAN, COBOL; estos tres últimos en su versión estructurada. Para programación orientada a objetos: C++, SMALLTALK y JAVA. Para programación simbólica: LISP, y para programación lógica: PROLOG. Cabe señalar que sólo hicimos mención de algunos lenguajes y tipos de lenguajes.

En este libro nos enfocaremos sobre el tipo de lenguajes de *programación estructurada*. En este enfoque los programas se diseñan de arriba hacia abajo (top–down) jerárquicamente, usando sólo un conjunto restringido de estructuras de control en cada nivel, instrucciones secuenciales, estructuras selectivas y estructuras repetitivas. Cuando esto se hace en forma adecuada el programa resulta muy fácil de entender, depurar y modificar.

Cuando tenemos que resolver un problema de tipo algorítmico, entendiendo por esto aquellos que tienen una solución determinística, primero desarrollamos el *algoritmo*, que proporciona una solución muy general. Posteriormente construimos el *diagrama de flujo*, que esquematiza gráfica y detalladamente la solución del problema, y a partir de éste, escribimos el *programa* en algún lenguaje de programación estructurado.

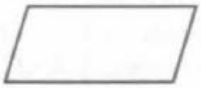
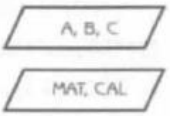
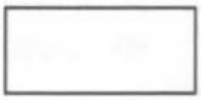
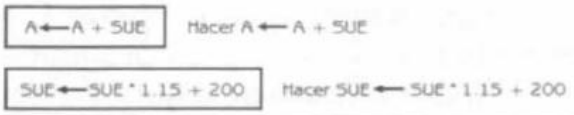
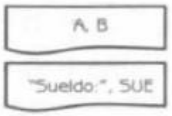
Pensamos que la tarea intelectual, la que requiere de un pensamiento profundo, de una capacidad de razonamiento flexible y crítica, es la de la construcción del diagrama de flujo, que representa la solución detallada del problema. La escritura del programa puede ser muy simple, conociendo las reglas sintácticas y semánticas que constituyen el lenguaje de programación.

El lenguaje que utilizaremos en este libro para la construcción de los programas es un *lenguaje algorítmico de pseudo-código*, independiente de cualquier lenguaje de programación. Esta característica consideramos que es muy importante, ya que permite al lector comprender las estructuras de datos y los algoritmos asociados a ellas sin relacionarlos a un lenguaje de programación en particular. Se considera que una vez que el lector domine correctamente estos conceptos, *muy fácilmente* podrá transportar los programas realizados a un lenguaje de programación, tal como PASCAL o C.

www.FreeLibros.com

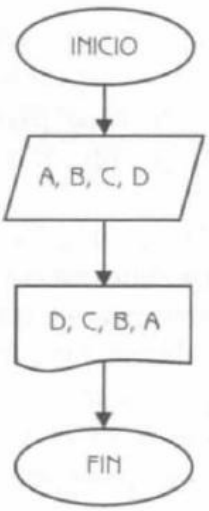
A continuación en la tabla 1.13 podemos observar los símbolos que utilizamos para la construcción del diagrama de flujo, la instrucción adecuada en el lenguaje algorítmico y un ejemplo de su uso. Los símbolos y las instrucciones que presentaremos son las que usaremos en una etapa inicial. Posteriormente, conforme avancemos en los temas de los siguientes capítulos, iremos presentando los símbolos restantes y las instrucciones correspondientes que permiten describirlos.

Tabla 1.13

Representación del símbolo	Instrucción	Ejemplo
	Leer...	
	Hacer...	
	Escribir...	

Ejemplo 1.11

Consideremos el diagrama de flujo 1.6, el mismo es idéntico al diagrama de flujo 1.1 construido para resolver el problema del ejemplo 1.6. A la derecha del diagrama podemos observar las instrucciones escritas en lenguaje algorítmico.



Instrucciones en lenguaje algorítmico

1. Leer A, B, C, D

2. Escribir D, C, B, A

Diagrama de Flujo 1.6

www.FreeLibros.com

Todo programa tiene un *nombre* que lo define. Utilizaremos las reglas presentadas para la construcción de un *identificador*, para construir el nombre del programa. Es muy importante que los *nombres de los programas* sean representativos de la función que cumplen los mismos. Posterior al nombre, debajo, es muy recomendable escribir un párrafo como comentario de lo que realiza el programa. También es necesario definir las constantes y variables que utilizaremos en el desarrollo de la solución.

A continuación presentamos el programa escrito en lenguaje algorítmico, del diagrama de flujo 1.6.

Programa 1.1

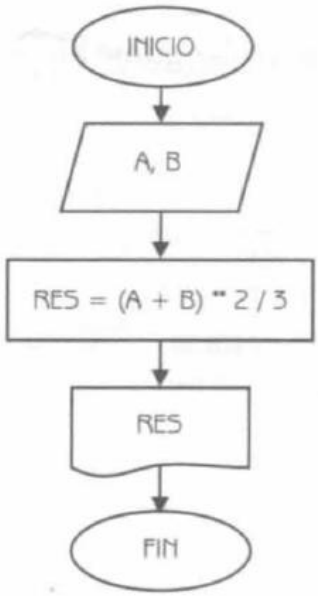
```
INVIERTE_DATOS

{El programa dado un conjunto de datos de entrada
invierte el orden de los mismos cuando los imprime}

{A, B, C y D son variables de tipo entero}

1. Leer A, B, C, D
2. Escribir D, C, B, A
```

Observe el lector que escribir un programa es muy sencillo, conociendo las instrucciones correspondientes. La tarea intelectual, creativa, radica en la construcción del diagrama de flujo.

www.FreeLibros.com

Ejemplo 1.12

Consideremos el diagrama de flujo 1.7, el mismo es idéntico al diagrama de flujo 1.2, construido para resolver el problema del ejemplo 1.7. A la derecha presentamos las instrucciones en lenguaje algorítmico.



Instrucciones en lenguaje algorítmico

1. Leer A, B
2. Hacer RES ← (A + B) ** 2 / 3
3. Escribir RES

Diagrama de Flujo 1.7

A continuación presentamos el programa en lenguaje algorítmico.

Programa 1.2

```
CALCULA

{El programa, dados como datos los enteros A y B,
calcula el resultado de una expresión}

{A y B son variables de tipo entero. RES es una variable
de tipo real}

1. Leer A, B
2. Hacer RES ← (A + B) ** 2 / 3
3. Escribir RES
```

www.FreeLibros.com

Ejemplo 1.13

En este ejemplo presentamos la solución en lenguaje algorítmico del diagrama de flujo 1.3.

Programa 1.3

```
PROMEDIO_CALIFICACION

{El algoritmo, dadas cinco calificaciones de un
alumno, calcula su promedio}

{MAT es una variable de tipo entero. CAL1, CAL2,
CAL3, CAL4, CAL5 y PRO son variables de tipo real}

1. Leer MAT, CAL1, CAL2, CAL3, CAL4, CAL5
2. Hacer PRO ←— (CAL1 + CAL2 + CAL3 + CAL4 + CAL5) / 5
3. Escribir MAT, PRO
```

Ejemplo 1.14

A continuación presentamos la solución en lenguaje algorítmico del diagrama de flujo 1.4.

Programa 1.4

```
CUADRADO_CUBO

{El programa, dado como dato un número entero
positivo, calcula el cuadrado y el cubo de dicho número}

{NUM es una variable de tipo entero. CUA y CUB son
variables de tipo real}

1. Leer NUM
2. Hacer CUA ←— NUM * NUM y CUB ←— NUM ** 3
3. Escribir CUA y CUB
```

www.FreeLibros.com

Ejemplo 1.15

En este ejemplo presentamos la solución en lenguaje algorítmico del diagrama de flujo 1.5.

Programa 1.5

```
PERIMETRO_SUPERFICIE_RECTANGULO

{El programa, dado como datos la base y la altura de
un rectángulo, calcula su perímetro y superficie}

{BASE, ALTU, SUP y PER son variables de tipo real}

1. Leer BASE, ALTU
2. Hacer SUP ⟵ BASE * ALTU y PER ⟵ 2 * (BASE + ALTU)
3. Escribir SUP y PER
```

A continuación presentamos una serie de *problemas resueltos*, diseñados expresamente como elementos de ayuda para el análisis y la retención de los conceptos. Además se utilizan en muchos de ellos, tablas que permiten hacer el seguimiento de la solución planteada en estos algoritmos.

www.FreeLibros.com

Problemas resueltos

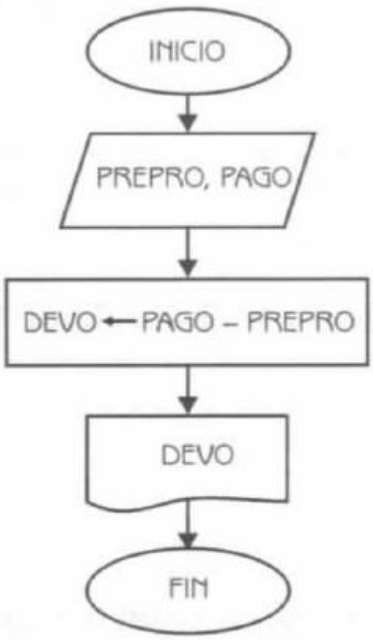
Problema 1.1

Construya un diagrama de flujo tal que dado el costo de un artículo vendido y la cantidad de dinero entregada por el cliente, calcule e imprima el cambio que se debe entregar al mismo.

Datos: PREPRO, PAGO

Donde:

PREPRO es una variable de tipo real que representa el precio del producto.
PAGO es una variable de tipo real que representa el pago que realiza el cliente.



Nota:
Asumimos que el pago del cliente es mayor al precio del producto.

Diagrama de Flujo 1.8

Explicación de las variables

DEVO: Variable de tipo real. Almacena el cambio que se debe entregar al cliente.

A continuación en la siguiente tabla, presentamos el seguimiento del algoritmo para diferentes corridas.

www.FreeLibros.com

Tabla 1.14			
NUMERO DE CORRIDA	DATOS		RESULTADO
	PREPRO	PAGO	DEVO
1	86.25	100	13.75
2	4.86	50	45.14
3	21.73	50	28.27
4	1.68	5	3.32
5	49.20	100	50.80

: Expresa valores que se imprimen

Programa 1.6

```
VUELTO_DE_UN_PAGO

{El programa, dado el costo de un producto y la cantidad de dinero entregada por el cliente, calcula el vuelto que hay que entregarle al mismo}

{PREPRO, PAGO y DEVO son variables de tipo real}

1. Leer PREPRO y PAGO
2. Hacer DEVO ←— PAGO – PREPRO
3. Escribir DEVO
```

Problema 1.2

Construya un diagrama de flujo tal que dadas la base y la altura de un triángulo, calcule e imprima su superficie.

Datos: BASE, ALTU

Donde:

BASE es una variable de tipo real que indica la base del triángulo.
ALTU es una variable de tipo real que representa la altura del triángulo.

Consideraciones:

La superficie de un triángulo se calcula aplicando la siguiente fórmula:

$$\frac{BASE * ALTU}{2}$$

Fórmula 1.3

www.FreeLibros.com

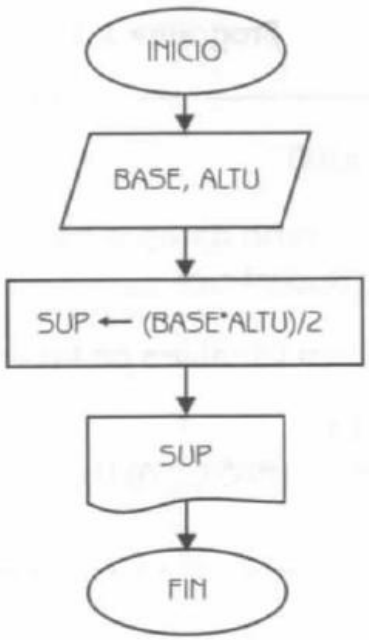


Diagrama de Flujo 1.9

Explicación de las variables

BASE y *ALTU*: Variables de tipo real.

SUP: Variable de tipo real. Almacena la superficie del triángulo.

A continuación en la siguiente tabla, presentamos el seguimiento del algoritmo para diferentes corridas.

Tabla 1.15

NUMERO DE CORRIDA	DATOS		RESULTADO
	BASE	ALTU	SUP
1	8.50	7.20	30.60
2	9.30	12.50	58.12
3	120.60	85.90	5179.77
4	4.33	1.98	4.28
5	11.60	7.40	42.92

: Expresa valores que se imprimen

A continuación presentamos el diagrama de flujo escrito en lenguaje algorítmico.

www.FreeLibros.com

Programa 1.7

```
SUPERFICIE_TRIANGULO

{El programa, dados como datos la base y la altura de un
triángulo, calcula su superficie}

{BASE, ALTU y SUP son variables de tipo real}

1. Leer BASE y ALTU
2. Hacer SUP ←— (BASE * ALTU) / 2
3. Escribir SUP
```

Problema 1.3

Escriba un diagrama de flujo tal que dado como datos el nombre de un dinosau rio, su peso y su longitud, expresados estos dos últimos en libras y pies respectiva mente; escriba el nombre del dinosaurio, su peso expresado en kilogramos y s longitud expresada en metros.

Datos: NOM, PES, LON

Donde:

NOM es una variable de tipo cadena de caracteres que indica el nombr del dinosaurio.

PES es una variable de tipo real que representa el peso del dinosauri en libras.

LON es una variable de tipo real que expresa la longitud del dinosauri en pies.

Consideraciones:

- 1 tonelada equivale a 1000 kilogramos.
- 1 pie equivale a 0.3047 metros.

www.FreeLibros.com

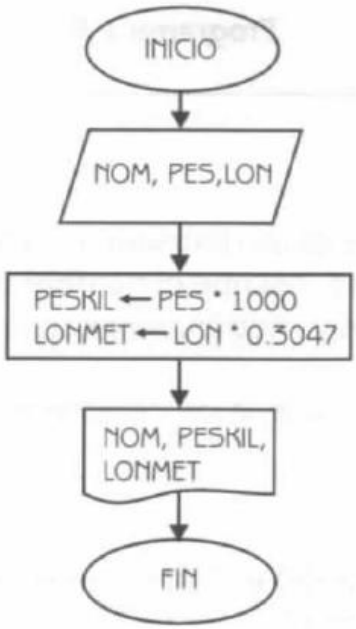


Diagrama de Flujo 1.10

Explicación de las variables

NOM: Variable de tipo cadena de caracteres.

PES y LON: Variables de tipo real.

PESKIL: Variable de tipo real. Almacena el peso del dinosaurio en kilogramos.

LONMET: Variable de tipo real. Almacena la longitud del dinosaurio en metros.

A continuación, en la tabla 1.16 podemos observar el seguimiento del algoritmo para diferentes corridas.

Tabla 1.16

NUMERO DE CORRIDA	DATOS			RESULTADOS	
	NOM	PES	LON	PESKIL	LONMET
1	PLATEOSAURUS	5	30	5000	9.14
2	DIPLOJOCUS	15	90	15000	27.42
3	BRACHIOSAURUS	50	80	50000	24.37
4	BRONTOSAURUS	25	70	25000	21.32
5	TYRANNOSAURUS	8	30	8000	9.14

: Expresa valores que se imprimen

www.FreeLibros.com

Programa 1.8

```
DINOSAURIOS

{El programa, dado el nombre de un dinosaurio, su peso expresado en toneladas y
su longitud expresada en libras; escribe el nombre del dinosaurio, su peso y longi-
tud expresadas en kilogramos y metros, respectivamente}

{NOM es una variable de tipo cadena de caracteres. PES, ALT, PESKIL y LONMET
son variables de tipo real}

1. Leer NOM, PES y LON
2. Hacer PESKIL ← PES * 1000 y LONMET ← LON * 0.3047
3. Escribir NOM, PESKIL y LONMET
```

Problema 1.4

Construya un diagrama de flujo que resuelva el problema que tienen en una gasolinera. Los surtidores de la misma registran lo que "surten" en galones, pero el precio de la gasolina está fijado en litros. El diagrama de flujo debe calcular e imprimir lo que hay que cobrarle al cliente.

Dato: GAL (variable de tipo real que representa los galones de gasolina que le surtieron a un cliente).

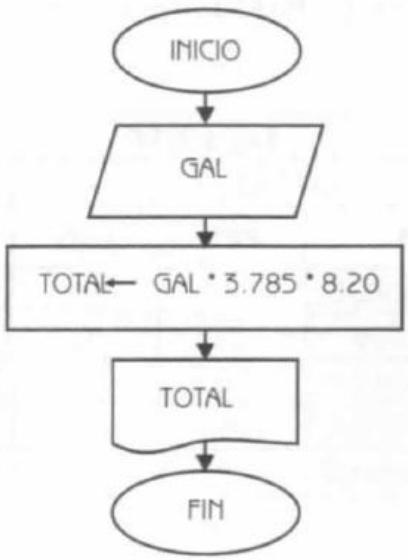


Diagrama de Flujo 1.11

www.FreeLibros.com

Consideraciones:

- Cada galón tiene 3.785 litros.
- El precio del litro es $8.20.

Explicación de las variables

GAL: Variable de tipo real.

TOTAL: Variable de tipo real. Almacena el total de lo que debe pagar el cliente.

A continuación, en la tabla 1.17 el lector podrá observar el seguimiento del algoritmo para diferentes corridas.

Tabla 1.17

NUMERO DE CORRIDA	DATOS	RESULTADO
	GAL	TOTAL
1	10.38	322.16
2	15.90	493.49
3	8.40	260.71
4	9.66	299.81
5	19.90	617.64

: Expresa valores que se imprimen

A continuación presentamos el diagrama de flujo, en lenguaje algorítmico.

Programa 1.9

```
GASOLINERA

{El programa, dado como dato los galones surtidos a un
cliente en una gasolinera, calcula lo que el mismo debe
pagar}

{GAL y TOTAL son variables de tipo real}

1. Leer GAL
2. Hacer TOTAL ←—— GAL * 3.785 * 8.20
3. Escribir TOTAL
```

www.FreeLibros.com

Problema 1.5

Construya un diagrama de flujo tal que dado como datos el radio y la altura de un cilindro, calcule e imprima el área y su volumen.

Datos: RADIO, ALTU

Donde:

RADIO es una variable de tipo real que representa el radio de un cilindro.
ALTU es una variable de tipo real que representa la altura del cilindro.

Consideraciones:

- El volumen de un cilindro lo calculamos aplicando la siguiente fórmula:

$$\text{Volumen} = \pi * \text{radio}^2 * \text{altura}$$

Fórmula 1.4

Donde: $\pi = 3.141592$.

- La superficie del cilindro la calculamos como:

$$\text{Area} = 2 * \pi * \text{radio} * \text{altura}$$

Fórmula 1.5

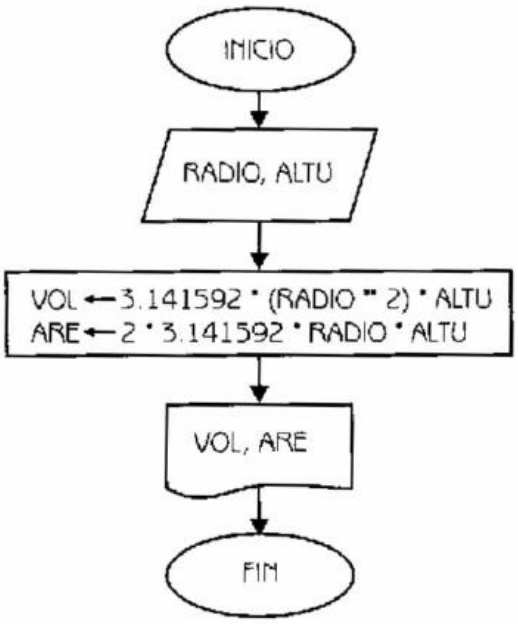


Diagrama de Flujo 1.12

www.FreeLibros.com

Explicación de las variables

RADIO y *ALTU*: Variables de tipo real.
VOL: Variable de tipo real. Almacena el volumen del cilindro.
ARE: Variable de tipo real. Almacena el área del cilindro.

En la tabla 1.18, podemos observar el seguimiento del algoritmo para diferentes corridas.

Tabla 1.18

NUMERO DE CORRIDA	DATOS		RESULTADOS	
	RADIO	ALTU	VOL	ARE
1	45.22	11.60	74519.33	3295.86
2	17.30	8.45	7945.09	918.51
3	69.30	72.40	1092332.40	31524.75
4	125.30	117.40	5790552.70	92427.01
5	85.90	237.20	5498583.10	128022.89

: Expresa valores que se imprimen

Programa 1.10

```
VOLUMEN_AREA_CILINDRO

{El programa dado como datos el radio y la altura de un
cilindro, calcula su área y su volumen.}

{RADIO, ALTU, VOL y ARE son variables de tipo real}

1. Leer RADIO y ALTU
2. Hacer VOL ←—— 3.141592 * (RADIO ** 2) * ALTU y
         ARE ←—— 2* 3.141592 * RADIO * ALTU
3. Escribir VOL y ARE
```

Problema 1.6

Construya un diagrama de flujo que calcule e imprima el número de segundos que hay en un determinado número de días.

Datos: DIAS (variable de tipo entero que expresa el número de días).

www.FreeLibros.com

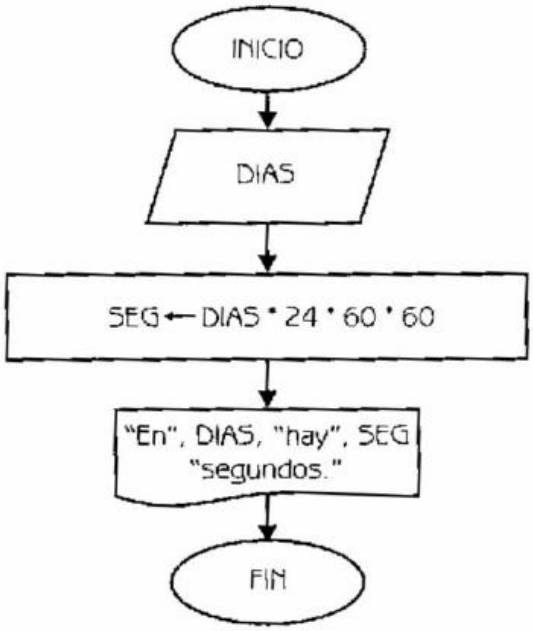


Diagrama de Flujo 1.13

Explicación de las variables

DIAS: Variable de tipo entero.

SEG: Variable de tipo real. Almacena la cantidad de segundos que hay en un número determinado de días.

Tabla 1.19

NUMERO DE CORRIDA	DATO	RESULTADO
	DIAS	SEG
1	7	604800
2	15	1296000
3	116	10022400
4	28	2419200
5	3	259200

: Expresa valores que se imprimen

A continuación presentamos al diagrama de flujo 1.13, en lenguaje algorítmico.

www.FreeLibros.com

Programa 1.11

```
SEGUNDOS_EN_DIAS

{El programa, dado un número determinado de días, calcula
cuántos segundos tienen éstos}

{DIAS es una variable de tipo entero. SEG es una variable de
tipo real}

1. Leer DIAS
2. Hacer SEG ⟵ DIAS * 24 * 60 * 60
3. Escribir "En ", DIAS, "hay ", SEG, "segundos."
```

Problema 1.7

Construya un diagrama de flujo tal que dados los tres lados de un triángulo, pueda determinar su área. Esta la calculamos aplicando la siguiente fórmula:

$$\text{Area} = \sqrt{S * (S - L1) * (S - L2) * (S - L3)}$$

$$S = (L1 + L2 + L3) / 2$$

Fórmula 1.6

Datos: L1, L2, L3 (variables de tipo real que representan los tres lados).

www.FreeLibros.com

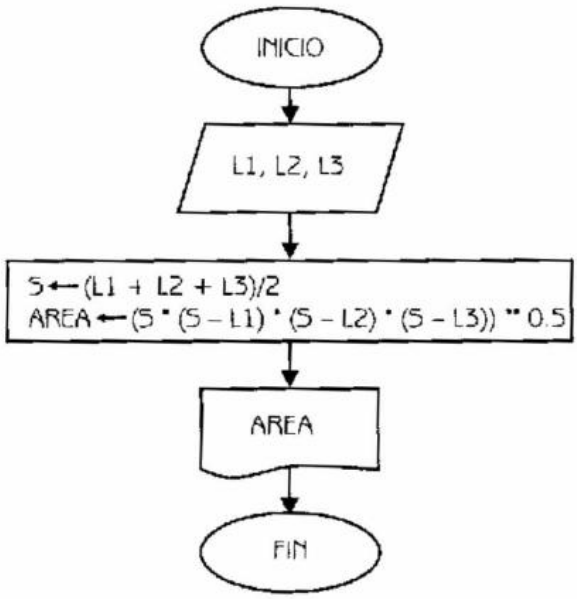


Diagrama de Flujo 1.14

Explicación de las variables

L1, *L2*, *L3*: Variables de tipo real.

S: Variable de tipo real. Se utiliza como una variable auxiliar para el cálculo del área.

AREA: Variable de tipo real. Almacena el área del triángulo.

En la tabla 1.20 podemos observar el seguimiento del algoritmo para diferentes corridas.

Tabla 1.20

NUMERO DE CORRIDA	DATOS			CALCULO AUXILIAR	RESULTADO
	L1	L2	L3	S	AREA
1	7.5	7.5	7.5	11.25	24.3569
2	6.1	4.8	3.4	7.15	8.1338
3	10.0	10.0	4.5	12.25	21.9230
4	2.0	16.0	15.8	16.90	15.7889
5	17.6	17.6	25.0	30.10	154.8739

: Expresa valores que se imprimen.

www.FreeLibros.com

Programa 1.12

```
AREA_TRIANGULO

{El programa, dados los tres lados de un triángulo, calcula su
área}

{L1, L2, L3, S y AREA son variables de tipo real.}

1. Leer L1, L2, L3
2. Hacer S ⟵ (L1 + L2 + L3) / 2 y
          AREA ⟵ (S * (S – L1) * (S – L2) * (S – L3)) ** 0.5
3. Escribir AREA
```

Problema 1.8

Construya un diagrama de flujo que calcule la distancia entre dos puntos, dado como datos las coordenadas de los puntos P1 y P2.

Datos: X1, Y1, X2, Y2

Donde:

X1 y Y1 son variables de tipo real que representan las coordenadas del punto P1 en el eje de las X y Y, respectivamente.

X2 y Y2 son variables de tipo real que indican las coordenadas del punto P2 en el eje de las X y Y.

Consideraciones:

- Para calcular la distancia "D" entre dos puntos dados P1 y P2 aplicamos la siguiente fórmula:

$$D = \sqrt{(X1 - X2)^2 + (Y1 - Y2)^2}$$

Fórmula 1.7

www.FreeLibros.com

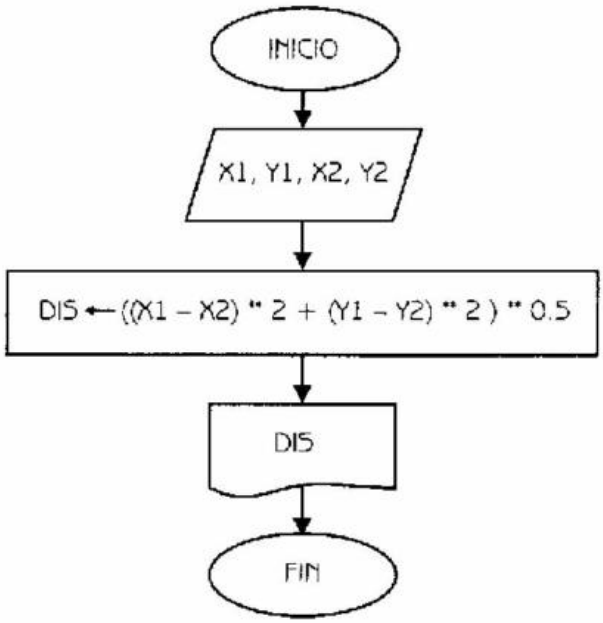


Diagrama de Flujo 1.15

Explicación de las variables

X1, Y1, X2, Y2: Variables de tipo real.

DIS: Variable de tipo real. Almacena la distancia entre dos puntos P1 y P2.

En la tabla 1.21 mostramos el seguimiento del algoritmo.

Tabla 1.21

NUMERO DE CORRIDA	DATOS				RESULTADO
	X1	Y1	X2	Y2	DIS
1	3.17	4.78	4.99	7.88	3.59
2	7.15	21.60	1.93	4.38	17.99
3	12.17	10.40	10.40	29.30	18.98
4	39.40	78.90	68.30	187.20	112.08
5	88.70	118.30	295.30	18.40	229.48

: Expresa valores que se imprimen

www.FreeLibros.com

Programa 1.13

```
DISTANCIA_ENTRE_DOS_PUNTOS

{El programa, dadas las coordenadas de dos puntos P1 y P2,
calcula la distancia entre estos puntos}

{X1, Y1, X2, Y2 y DIS son variables de tipo real}

1. Leer X1, Y1, X2, Y2
2. Hacer DIS ←— ((X1 – X2) ** 2 + (Y1 – Y2) ** 2) ** 0.5
3. Escribir DIS
```

www.FreeLibros.com

2

Estructuras algorítmicas *selectivas*

2.1 Introducción

Las estructuras lógicas selectivas se encuentran en la solución algorítmica de casi todo tipo de problemas. Las utilizamos cuando en el desarrollo de la solución de un problema debemos *tomar una decisión*, para establecer un proceso o señalar un camino alternativo a seguir.

Esta toma de decisión (expresada en el diagrama de flujo con un rombo) se basa en la evaluación de una o más condiciones que nos señalarán como alternativa o consecuencia, la *rama* a seguir.

Hay situaciones en las que la toma de decisiones se realiza en cascada. Es decir se toma una decisión, se marca la rama correspondiente a seguir, se vuelve a tomar otra decisión y así sucesivamente. Por lo que para alcanzar la solución de un problema o subproblema debemos aplicar prácticamente un árbol de decisión.

Las estructuras algorítmicas selectivas que se utilizan para la toma de decisiones lógicas las podemos clasificar de la siguiente forma:

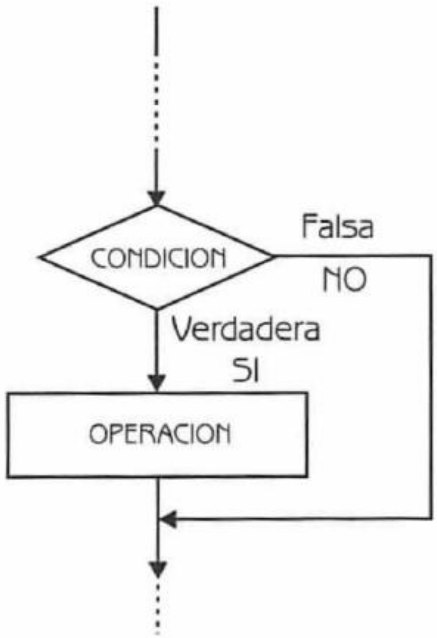
1. SI ENTONCES (Estructura selectiva simple)
2. SI ENTONCES / SINO (Estructura selectiva doble)
3. SI MULTIPLE (Estructura selectiva múltiple)

Cabe señalar que cuando a las estructuras selectivas las aplicamos en cascada, podemos utilizar una combinación de las estructuras señaladas anteriormente en la clasificación.

www.FreeLibros.com

2.2 La estructura selectiva simple *si entonces*

La estructura selectiva *si entonces* permite que el flujo del diagrama siga por un camino específico si se cumple una condición o conjunto de condiciones. Si al evaluar la condición (o condiciones) el resultado es verdadero, entonces se ejecuta(n) cierta(s) operación(es). Luego se continúa con la secuencia normal del diagrama (diagrama de flujo 2.1).



Donde:
CONDICION expresa la condición o conjunto de condiciones a evaluar.
OPERACION expresa la operación o conjunto de operaciones que se van a realizar si la condición resulta verdadera.

Diagrama de Flujo 2.1

A continuación mostramos el diagrama de flujo 2.1, que ilustra la estructura selectiva *si entonces*, en lenguaje algorítmico.

Programa 2.1

```
.
.
.
Si condición entonces
        Hacer operación
{Fin del condicional}
.
.
.
```

www.FreeLibros.com

Ejemplo 2.1

Construya un diagrama de flujo tal que dado como dato la calificación de un alumno en un examen, escriba "aprobado" en caso de que esa calificación sea mayor a 8.

Dato: CAL (variable de tipo real que representa la calificación del alumno).

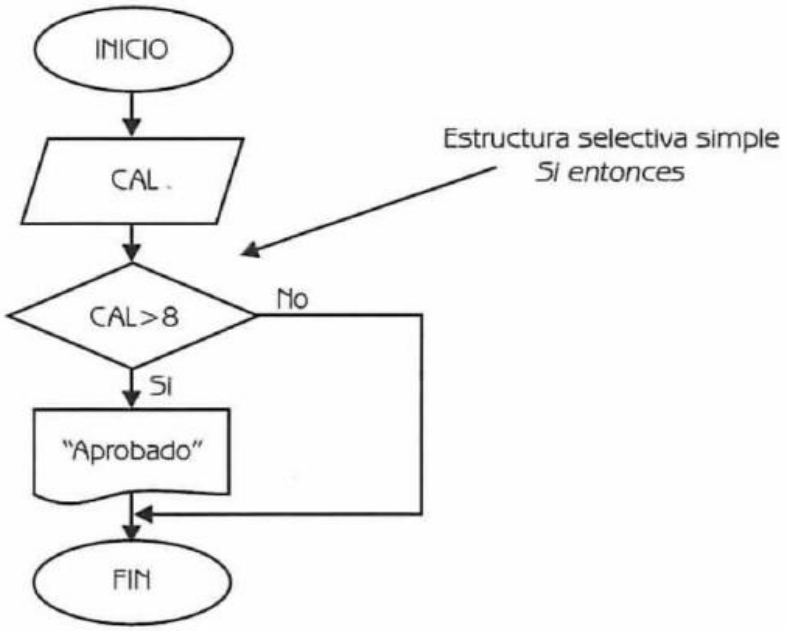


Diagrama de Flujo 2.2

En la tabla 2.1, podemos observar el seguimiento del diagrama de flujo para diferentes corridas.

Tabla 2.1

NUMERO DE CORRIDA	DATO	RESULTADO
	CAL	
1	8.75	"Aprobado"
2	7.90	
3	8.00	
4	9.50	"Aprobado"
5	8.35	"Aprobado"

: Expresa valores que se imprimen.

A continuación presentamos el diagrama de flujo en lenguaje algorítmico.

www.FreeLibros.com

Programa 2.2

```
EXAMEN_SELECTIVA_SIMPLE

{El programa, dado como dato la calificación de un alumno en un
examen, escribe aprobado si la calificación es superior a 8}

{CAL es una variable de tipo real}

1. Leer CAL
2. Si CAL > 8 entonces
       Escribir "Aprobado"
3. {Fin del condicional del paso 2}
```

Ejemplo 2.2

Dado como dato el sueldo de un trabajador, aplíquele un aumento del 15% si su sueldo es inferior a $1000. Imprima en este caso el nuevo sueldo del trabajador. Haga el diagrama de flujo correspondiente.

Dato: SUE (variable de tipo real que representa el sueldo del trabajador).

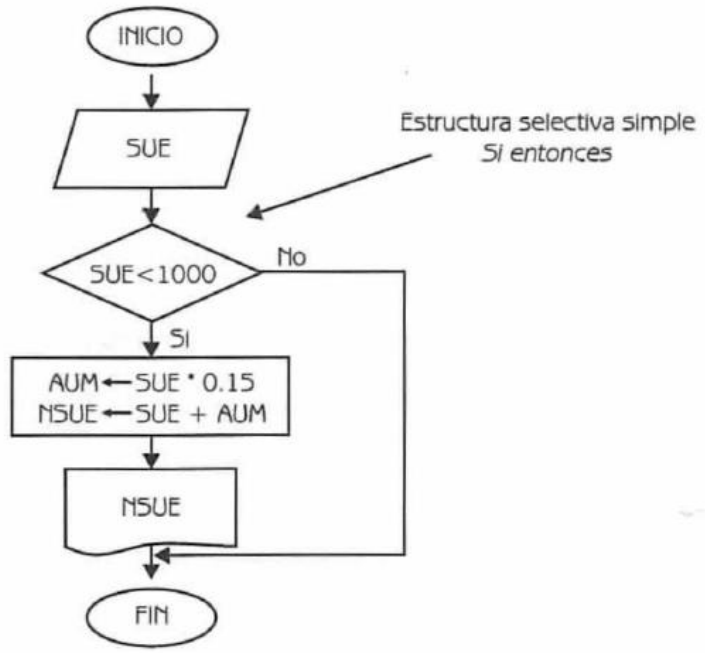


Diagrama de Flujo 2.3

Nota:
Las operaciones de asignación podríamos haberlas simplificado, realizando la siguiente operación:
NSUE ← SUE * 1.15

Explicación de las variables

SUE: Variable de tipo real.

www.FreeLibros.com

AUM: Variable de tipo real. Almacena el aumento del trabajador.
NSUE: Variable de tipo real. Almacena el nuevo sueldo del trabajador.

En la tabla 2.2, observamos el seguimiento del algoritmo para diferentes corridas.

Tabla 2.2

NUMERO DE CORRIDA	DATO	CALCULO AUXILIAR	RESULTADO
	SUE	AUM	NSUE
1	875.50	131.32	1006.82
2	1300.00		
3	2150.00		
4	976.00	146.40	1122.40
5	785.00	117.75	902.75
6	937.80	140.67	1078.47

: Expresa los valores que se imprimen.

A continuación presentamos el programa correspondiente.

Programa 2.3

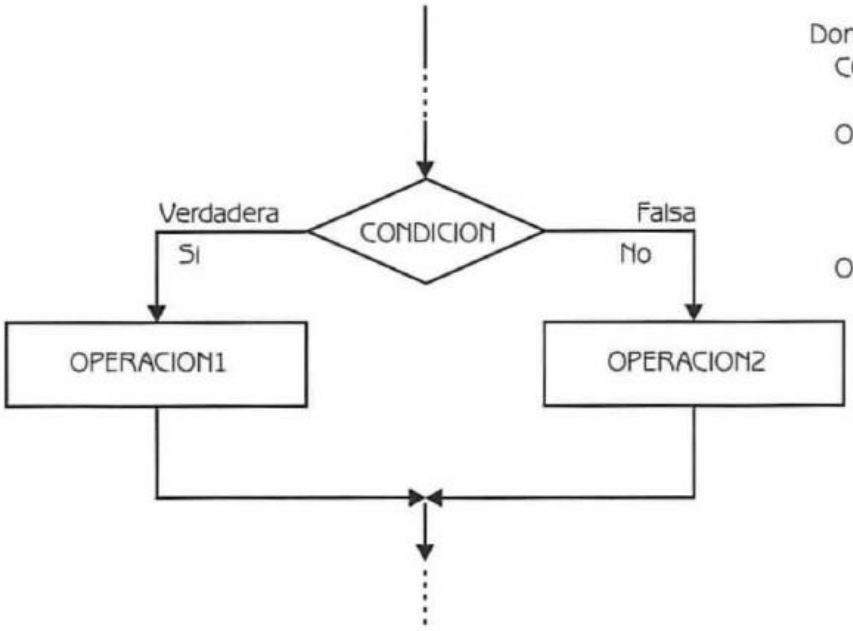
```
AUMENTO_SELECTIVA_SIMPLE

{El programa dado como dato el sueldo de un trabajador, le aplica
un aumento del 15% si su sueldo es inferior a $1000}

{SUM, AUM y NSUE son variables de tipo real}

1. Leer SUE
2. Si SUE < 1000 entonces
     Hacer AUM ←— SUE * 0.15 y NSUE ←— SUE + AUM
     Escribir NSUE
3. {Fin del condicional del paso 2}
```

2.3 La estructura selectiva doble *si entonces / sino*

La estructura selectiva *si entonces/sino* permite que el flujo del diagrama se bifurque por dos ramas diferentes en el punto de la toma de decisión(es). Si al evaluar la condición (o condiciones) el resultado es verdadero, entonces se sigue por un camino específico y se ejecuta(n) cierta(s) operación(es). Por otra parte, si el resultado es falso entonces se sigue por otro camino y se ejecuta(n) otra(s) opera-

www.FreeLibros.com

ción(es). En ambos casos, luego de ejecutarse la(s) operación(es) indicada(s), se continúa con la secuencia normal del diagrama. A continuación presentamos el diagrama de flujo 2.4 que ilustra esta estructura selectiva.



Donde:
CONDICION expresa la condición o conjunto de condiciones a evaluarse.
OPERACION1 expresa la operación o conjunto de operaciones que se van a realizar si la condición resulta verdadera.
OPERACION2 expresa la operación o conjunto de operaciones que se van a realizar si la condición resulta falsa.

Diagrama de Flujo 2.4

El diagrama de flujo en lenguaje algorítmico se representa de esta forma.

Programa 2.4

```
.
.
.
Si condición
    entonces
        Hacer operación 1
    sino
        Hacer operación 2
{Fin del condicional}
.
.
.
```

www.FreeLibros.com

Ejemplo 2.3

Construya un diagrama de flujo tal que dado como dato la calificación de un alumno en un examen, escriba "aprobado" si su calificación es mayor o igual que 8 y "reprobado" en caso contrario.

Dato: CAL (variable de tipo real que expresa la calificación del alumno).

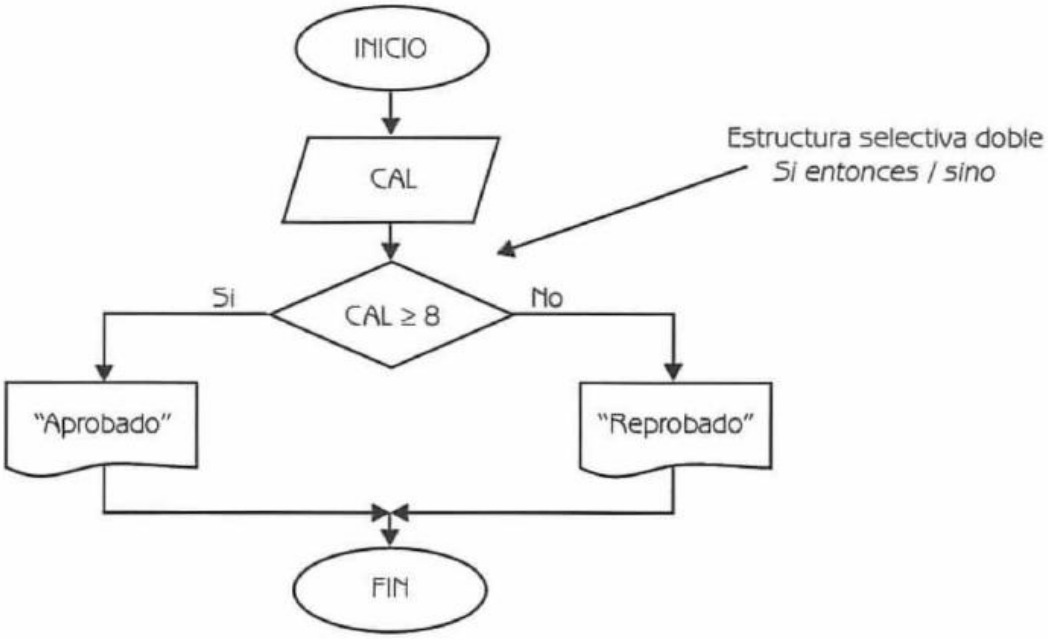


Diagrama de Flujo 2.5

En la tabla 2.3 podemos observar el seguimiento del algoritmo para diferentes corridas.

Tabla 2.3

NUMERO DE CORRIDA	DATO	RESULTADO
	CAL	
1	8.75	"Aprobado"
2	7.90	"Reprobado"
3	8.00	"Aprobado"
4	9.50	"Aprobado"
5	8.35	"Aprobado"

: Expresa valores que se imprimen.

A continuación presentamos el programa correspondiente.

www.FreeLibros.com

Programa 2.5

```
EXAMEN_SELECTIVA_DOBLE

{El programa, dado como dato la calificación de un alumno
en un examen, escribe "aprobado" si su calificación es mayor
o igual que 8 y "reprobado" en caso contrario}

{CAL es una variable de tipo real}

1. Leer CAL
2. Si CAL >= 8
        entonces
            Escribir "Aprobado"
        sino
            Escribir "Reprobado"
3. {Fin del condicional del paso 2}
```

Ejemplo 2.4

Construya un diagrama de flujo tal que dado como dato el sueldo de un trabajador, le aplique un aumento del 15% si su sueldo es inferior a $1000 y 12% en caso contrario. Imprima el nuevo sueldo del trabajador.

Dato: SUE (variable de tipo real que representa el sueldo del trabajador).

www.FreeLibros.com

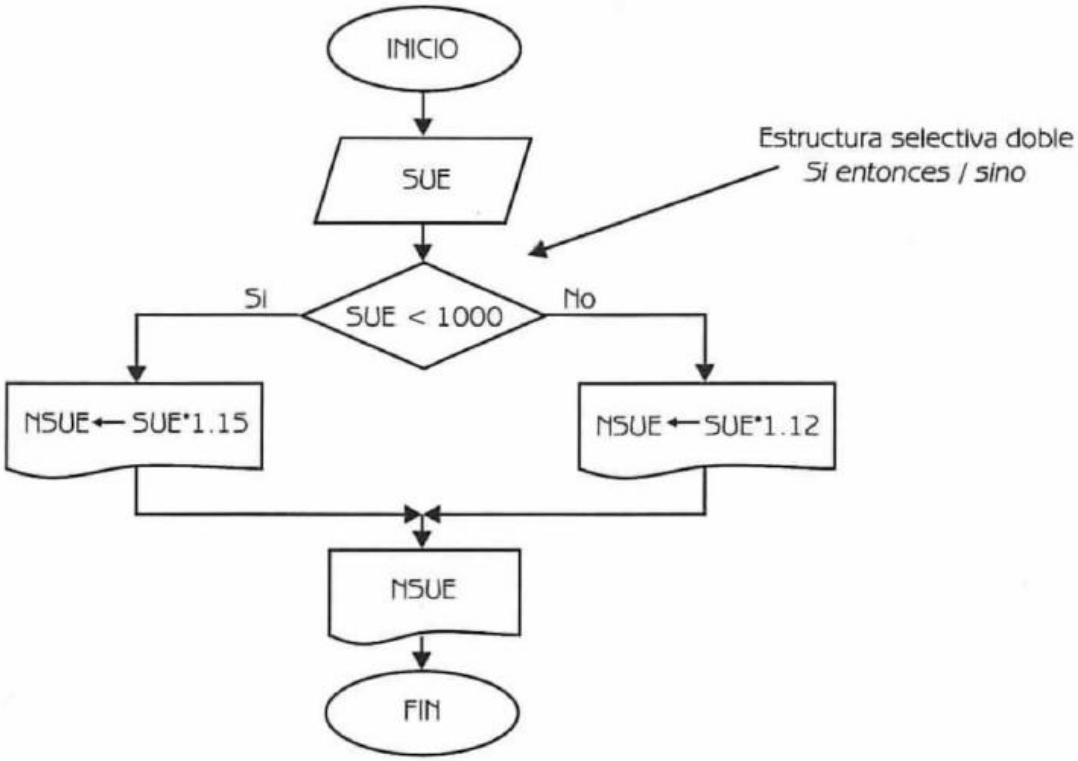


Diagrama de Flujo 2.6

Explicación de las variables

SUE: Variable de tipo real.

NSUE: Variable de tipo real. Almacena el nuevo sueldo del trabajador.

En la tabla 2.4 mostramos el seguimiento del algoritmo para diferentes corridas.

Tabla 2.4

NUMERO DE CORRIDA	DATO	RESULTADO
	SUE	NSUE
1	840.50	966.57
2	1200.00	1344.00
3	1950.00	2184.00
4	680.70	782.80
5	930.80	1070.42
6	1000.00	1120.00

: Expresa los valores que se imprimen.

www.FreeLibros.com

Programa 2.6

```
AUMENTO_SELECTIVA_DOBLE

{El programa dado como dato el sueldo de un trabajador, le
aplica un aumento del 15% si su sueldo es inferior a $1000 y
12% en caso contrario}

{SUE y NSUE son variables de tipo real}

1. Leer SUE
2. Si SUE < 1000
        entonces
            Hacer NSUE ←—— SUE * 1.15
        sino
            Hacer NSUE ←—— SUE * 1.12
3. {Fin del condicional del paso 2}
4. Escribir NSUE
```

2.4 La estructura selectiva múltiple *si múltiple*

La estructura selectiva *si múltiple* permite que el flujo del diagrama se bifurque por varias ramas en el punto de la toma de decisión(es), esto en función del valor que tome el selector. Así si el selector toma el *valor 1* se ejecutará la acción 1, si toma el *valor 2* se ejecutará la acción 2, si toma el *valor N* se realizará la acción N, y si toma un valor distinto de los valores comprendidos entre 1 y N, se continuará con el flujo normal del diagrama realizándose la acción N + 1.

A continuación presentamos el diagrama de flujo 2.7 que ilustra esta estructura selectiva.

Donde:

SELECTOR	es la variable o expresión a evaluarse, según la cual se tomará una de las "múltiples" decisiones o alternativas.
ACCION 1	expresa la operación o conjunto de operaciones que se van a realizar si el selector toma el *valor 1*.
ACCION 2	expresa la operación o conjunto de operaciones que se van a realizar si el selector toma el *valor 2*.
ACCION N	expresa la operación o conjunto de operaciones que se van a realizar si el selector toma el *valor N*.
ACCION N+1	expresa la operación que se va a realizar cuando se continúe con el flujo normal del diagrama.

www.FreeLibros.com

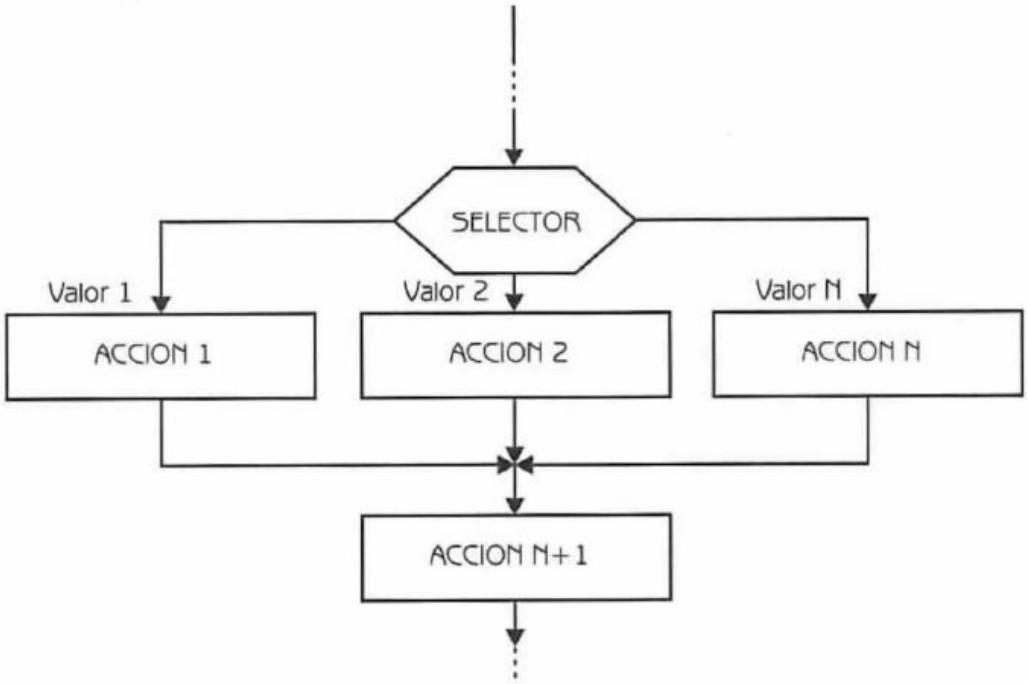


Diagrama de Flujo 2.7

El diagrama de flujo 2.7 en lenguaje algorítmico lo expresamos de esta forma.

Programa 2.7

```
.
.
.
Si selector igual
      Valor 1: Hacer acción 1
      Valor 2: Hacer acción 2
      .
      .
      .
      Valor N: Hacer acción N
{Fin del condicional}
Hacer acción N+1
.
.
.
```

La estructura selectiva *si múltiple* es muy flexible, lo que permite aplicarse de diferentes formas. Obsérvense los siguientes diagramas de flujo y las explicaciones correspondientes.

www.FreeLibros.com

Ejemplo 2.5

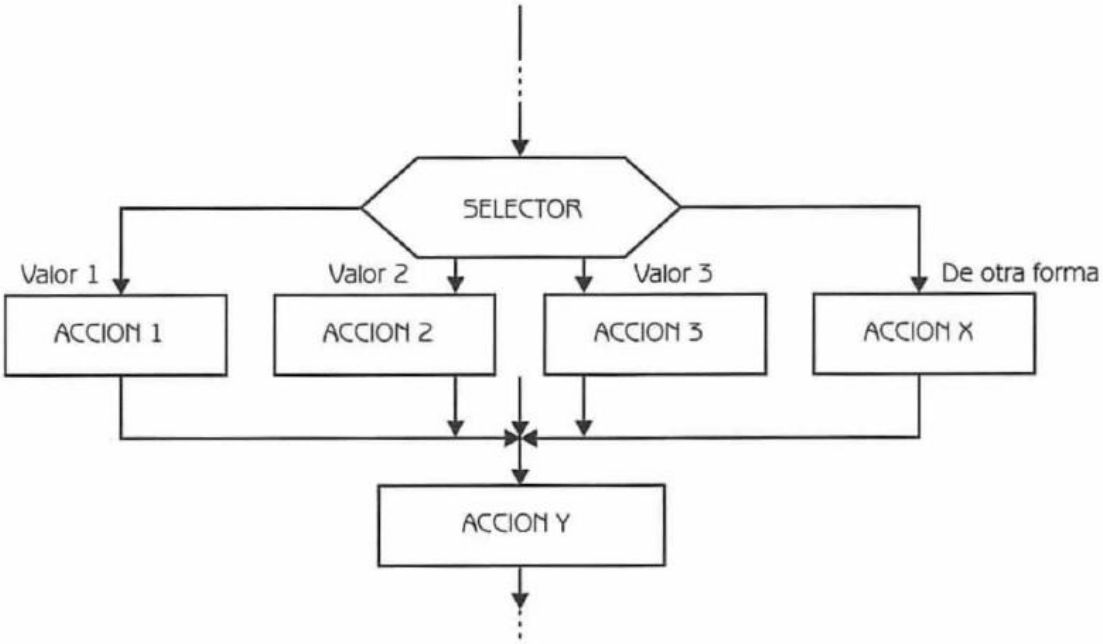


Diagrama de Flujo 2.8

Nota: Observe el lector que si el selector toma el *valor 1* se ejecuta la acción 1, si toma el *valor 2* se realiza la acción 2, si toma el *valor 3* se realiza la acción 3, y si toma cualquier otro valor se realiza la acción X. Luego cuando se continúa con el flujo normal del diagrama se realiza la acción Y.

El diagrama de flujo en lenguaje algorítmico, lo expresamos de esta forma:

Programa 2.8

```
.
.
.
Si selector igual
      Valor 1: Hacer acción 1
      Valor 2: Hacer acción 2
      Valor 3: Hacer acción 3
      De otra forma: Hacer acción X
{Fin del condicional}
Hacer acción Y
.
.
.
```

www.FreeLibros.com

Ejemplo 2.6

Analicemos el siguiente caso.

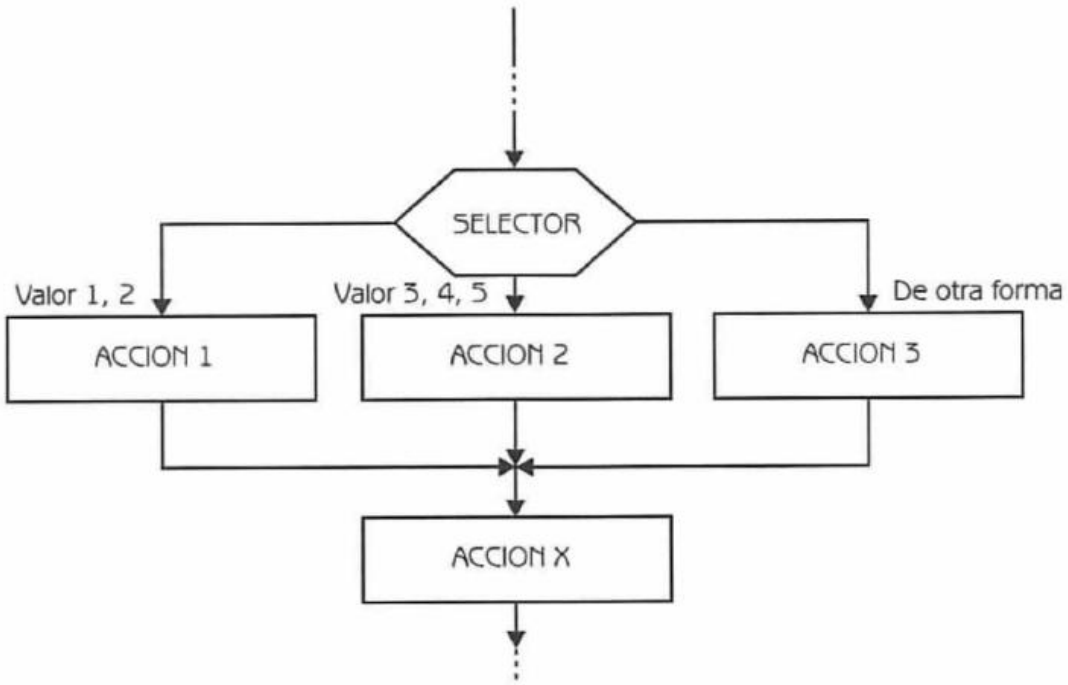


Diagrama de Flujo 2.9

Nota: Observe el lector que si el selector toma el *valor 1 o 2* se realiza la acción 1, si el selector toma el *valor 3, 4 o 5* se realiza la acción 2, y si el selector toma cualquier otro valor se realiza la acción 3. Luego, cuando se continúa con el flujo normal del diagrama se realiza la acción X.

A continuación presentamos al diagrama de flujo 2.9 en lenguaje algorítmico.

Programa 2.9

```
.
.
.
Si selector igual
      Valor 1,2: Hacer acción 1
      Valor 3,4,5: Hacer acción 2
      De otra forma: Hacer acción 3
{Fin del condicional}
Hacer acción X
.
.
.
```

www.FreeLibros.com

A continuación presentamos algunos ejemplos donde el lector puede aplicar los conceptos estudiados con la estructura selectiva *si múltiple*.

Ejemplo 2.7

Construya un diagrama de flujo tal que dado como datos dos variables de tipo entero, obtenga el resultado de la siguiente función:

$$VAL = \begin{cases} 100 * V & \text{Si NUM} = 1 \\ 100 ** V & \text{Si NUM} = 2 \\ 100 / V & \text{Si NUM} = 3 \\ 0 & \text{Para cualquier otro valor de NUM} \end{cases}$$

Datos: NUM, V

Donde:

NUM es una variable de tipo entero que representa el tipo (opción) de cálculo que se va a realizar.

V es una variable de tipo entero que se utiliza para el cálculo de la función.

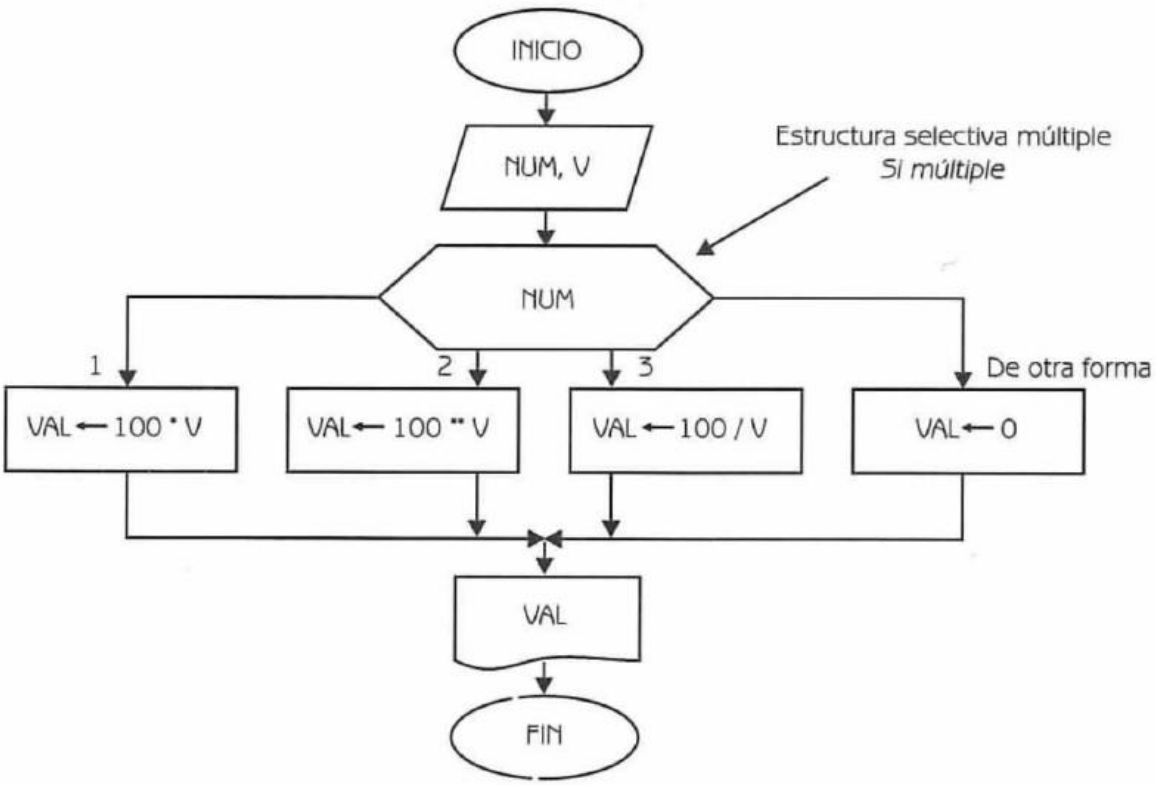


Diagrama de Flujo 2.10

www.FreeLibros.com

Explicación de las variables

NUM: Variable de tipo entero.

V: Variable de tipo entero.

VAL: Variable de tipo real. Almacena el resultado de la función.

En la tabla 2.5 podemos observar el seguimiento del algoritmo para diferentes corridas.

Tabla 2.5

NUMERO DE CORRIDA	DATOS		RESULTADO VAL
	NUM	V	
1	1	8	800
2	7	6	0
3	3	4	25
4	4	8	0
5	2	3	1000000
6	3	10	10

: Expresa los valores que se imprimen.

Programa 2.10

```
FUNCION_SELECTIVA_MULTIPLE

{El programa, dados como datos dos variables de tipo
entero, calcula el resultado de una función}

{NUM y V son variables de tipo entero. VAL es una variable de
tipo real}

1. Leer NUM y V
2. Si NUM igual
        1: Hacer VAL ←—— 100 * V
        2: Hacer VAL ←—— 100 ** V
        3: Hacer VAL ←—— 100 / V
        De otra forma: Hacer VAL ←—— 0
3. {Fin del condicional del paso 2}
4. Escribir VAL
```

www.FreeLibros.com

Ejemplo 2.8

Construya un diagrama de flujo tal que dado como datos la categoría y el sueldo de un trabajador, calcule el aumento correspondiente teniendo en cuenta la siguiente tabla. Imprima la categoría del trabajador y su nuevo sueldo.

Tabla 2.6

CATEGORÍA	AUMENTO
1	15%
2	10%
3	8%
4	7%

Datos: CATE, SUE

Donde:

CATE es una variable de tipo entero que representa la categoría del trabajador.
SUE es una variable de tipo real que expresa el sueldo del trabajador.

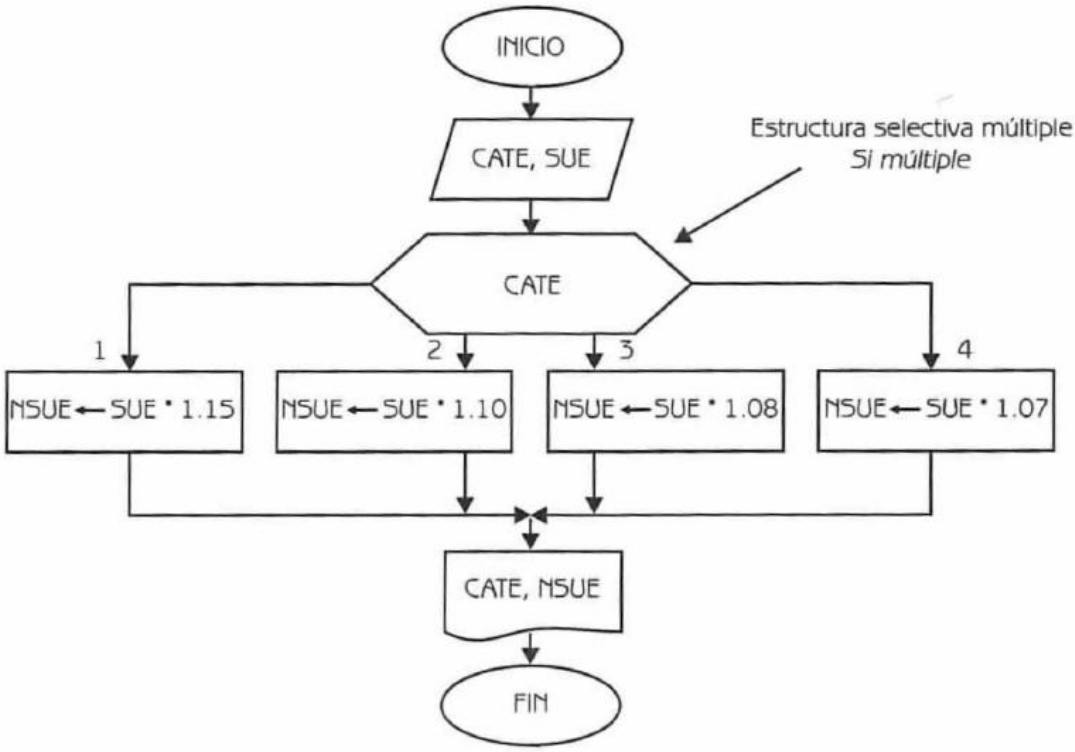


Diagrama de Flujo 2.11

www.FreeLibros.com

Explicación de las variables

CATE: Variable de tipo entero.

SUE: Variable de tipo real.

NSUE: Variable de tipo real. Almacena el sueldo con el aumento incorporado.

A continuación en la siguiente tabla podemos observar el seguimiento del algoritmo para diferentes corridas.

Tabla 2.7

NUMERO DE CORRIDA	DATOS		RESULTADO NSUE
	CATE	SUE	
1	3	3800	4104
2	4	6200	6634
3	1	1100	1265
4	2	1750	1925
5	3	4100	4428

: Expresa valores que se imprimen.

El programa en lenguaje algorítmico es el siguiente:

Programa 2.11

```
AUMENTO_SELECTIVA_MULTIPLE

{El programa calcula el aumento de sueldo de los trabajadores,
teniendo en cuenta su categoría. El aumento se incorpora al salario}

{CATE es una variable de tipo entero. SUE y NSUE son variables de
tipo real}

1. Leer CATE, SUE
2. Si CATE igual
        1: Hacer NSUE ←— SUE * 1.15
        2: Hacer NSUE ←— SUE * 1.10
        3: Hacer NSUE ←— SUE * 1.08
        4: Hacer NSUE ←— SUE * 1.07
3. {Fin del condicional del paso 2}
4. Escribir CATE, NSUE
```

www.FreeLibros.com

2.5 Estructuras selectivas en cascada (anidadas)

Encontramos numerosos casos en el desarrollo de la solución de problemas en el que luego de *tomar una decisión* y marcar el camino correspondiente a seguir, es necesario tomar otra decisión. Se señala, luego de evaluar las condiciones, la rama correspondiente a seguir, y nuevamente podemos tener que tomar otra decisión. El proceso puede repetirse numerosas veces. En este caso, para resolver el problema, estamos aplicando estructuras selectivas en cascada o anidadas.

A continuación en los siguientes ejemplos analizaremos casos diferentes.

Ejemplo 2.9

Analicemos el siguiente caso, donde dentro de la estructura selectiva *si entonces* encontramos la estructura selectiva *si entonces / sino*.

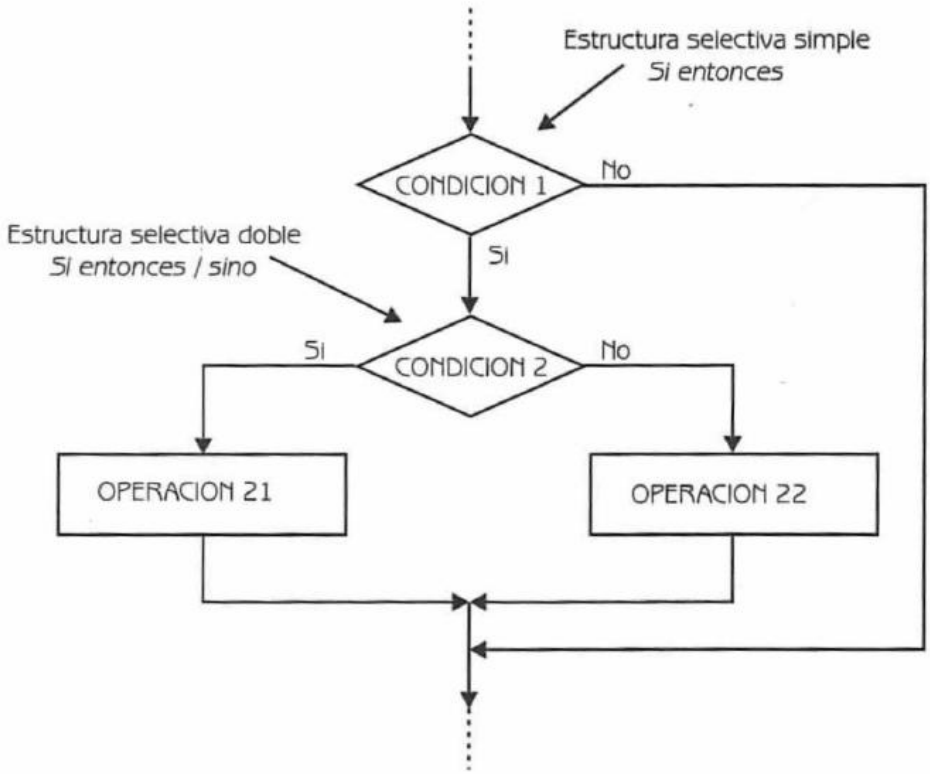


Diagrama de Flujo 2.12

A continuación presentamos el programa en lenguaje algorítmico.

www.FreeLibros.com

Programa 2.12

```
.
.
.
N.      Si condición1 entonces
          N.1.      Si condición2
                          entonces
                              Hacer operación21
                          sino
                              Hacer operación22
          N.2. {Fin del condicional del paso N.1}
N+1. {Fin del condicional del paso N}
.
.
.
```

Ejemplo 2.10

Analicemos el siguiente caso.

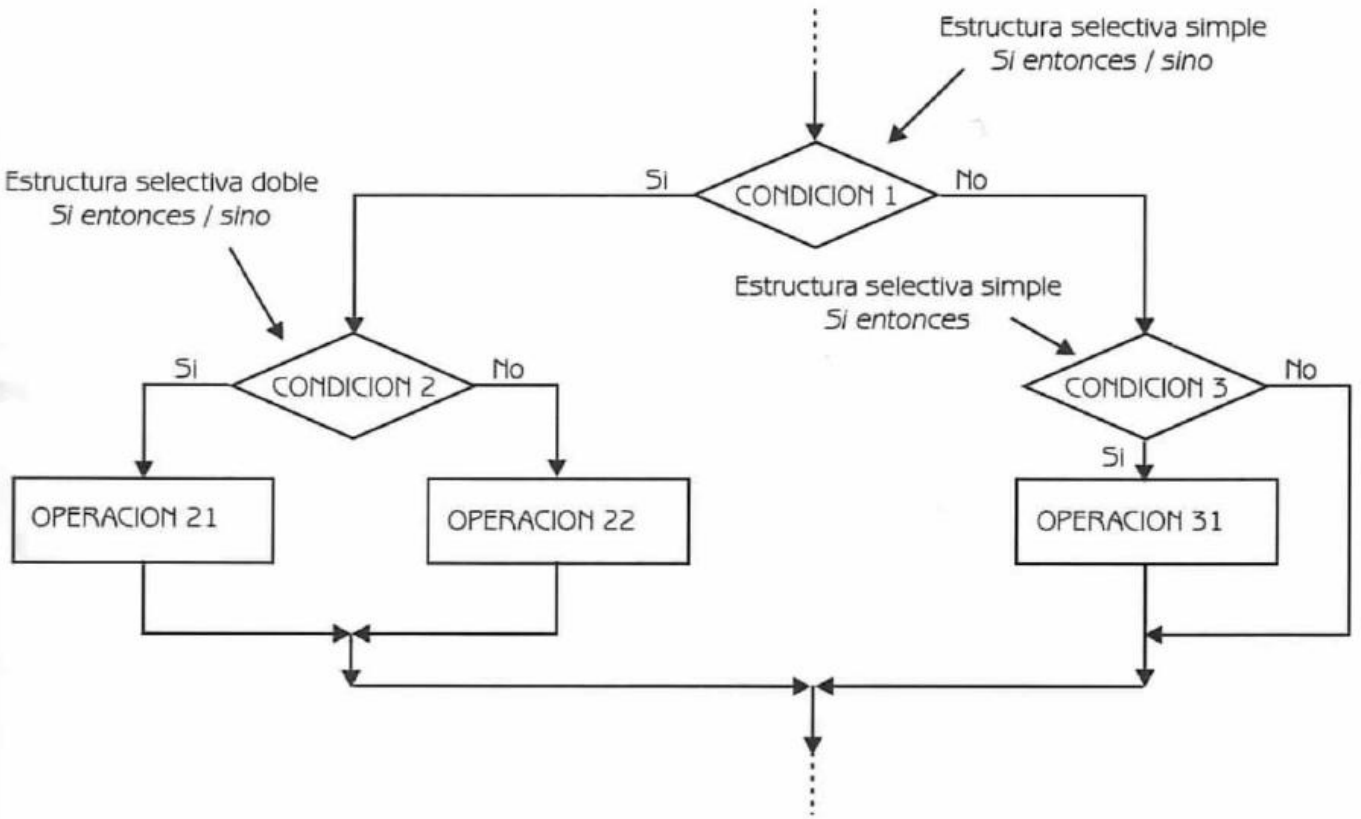


Diagrama de Flujo 2.13

www.FreeLibros.com

Nota: Obsérvese que dentro de la estructura selectiva *si entonces / sino* existen otras dos estructuras selectivas. Si al evaluar la condición 1, ésta resulta verdadera entonces tenemos que evaluar la condición 2 (estructura selectiva *si entonces / sino*). Por otra parte si la condición 1 resulta falsa, entonces tenemos que evaluar la condición 3 (estructura selectiva *si entonces*).

El diagrama de flujo en lenguaje algorítmico lo expresamos de esta forma.

Programa 2.13

```
.
.
.
N.   Si condición1
          entonces
          N.1.  Si condición2
                     entonces
                          Hacer operación21
                     sino
                          Hacer operación22
          N.2.  {Fin del condicional del paso N.1}
            sino
          N.3.  Si condición3 entonces
                          Hacer operación31
          N.4.  {Fin del condicional del paso N.3}
N+1. {Fin del condicional del paso N}
.
.
.
```

www.FreeLibros.com

Ejemplo 2.11

Estudiemos el último caso.

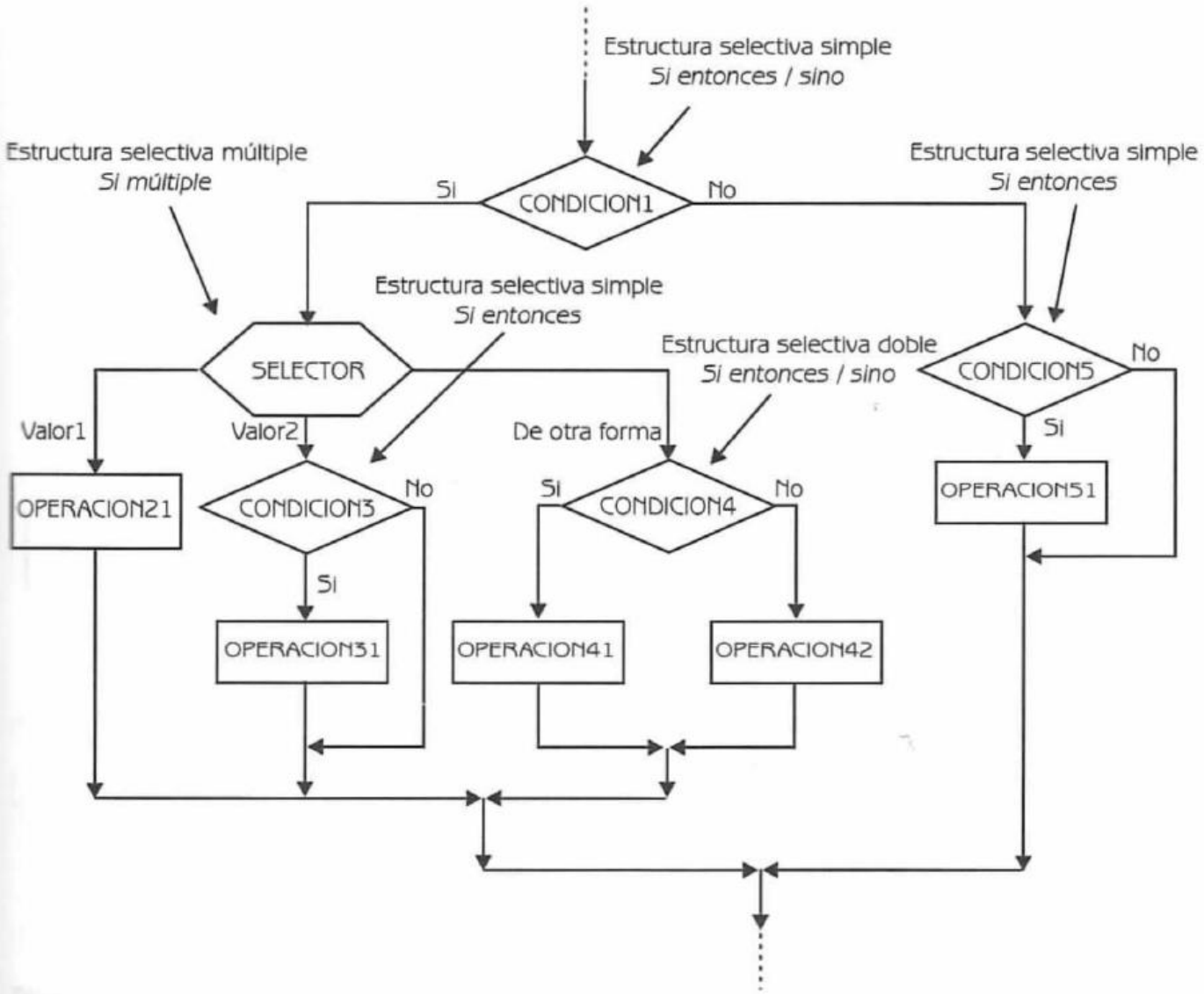


Diagrama de Flujo 2.14

Nota: Observe el lector que dentro de la estructura selectiva *si entonces / sino*, encontramos otras dos estructuras selectivas: *si múltiple* y *si entonces*. A su vez, dentro de la estructura selectiva *si múltiple* encontramos otras dos estructuras selectivas.

A continuación mostramos el programa correspondiente.

www.FreeLibros.com

Programa 2.14

```
.
.
.
N.  Si condición1
        entonces
        N.1.  Si selector1 igual
                  Valor 1: Hacer operación21
                  Valor 2:
                  N.1.1.    Si condición3 entonces
                                Hacer operación31
                  N.1.2.    {Fin del condicional del paso N.1.1}
                  De otra forma:
                  N.1.3.    Si condición4
                                entonces
                                    Hacer operación41
                                sino
                                    Hacer operación42
                  N.1.4.    {Fin del condicional del paso N.1.3}
        N.2.  {Fin de condicional del paso N.1}
        sino
        N.3.  Si condición5 entonces
                  Hacer operación51
        N.4.  {Fin del condicional del paso N.3}
N+1.{Fin del condicional del paso N}
.
.
.
```

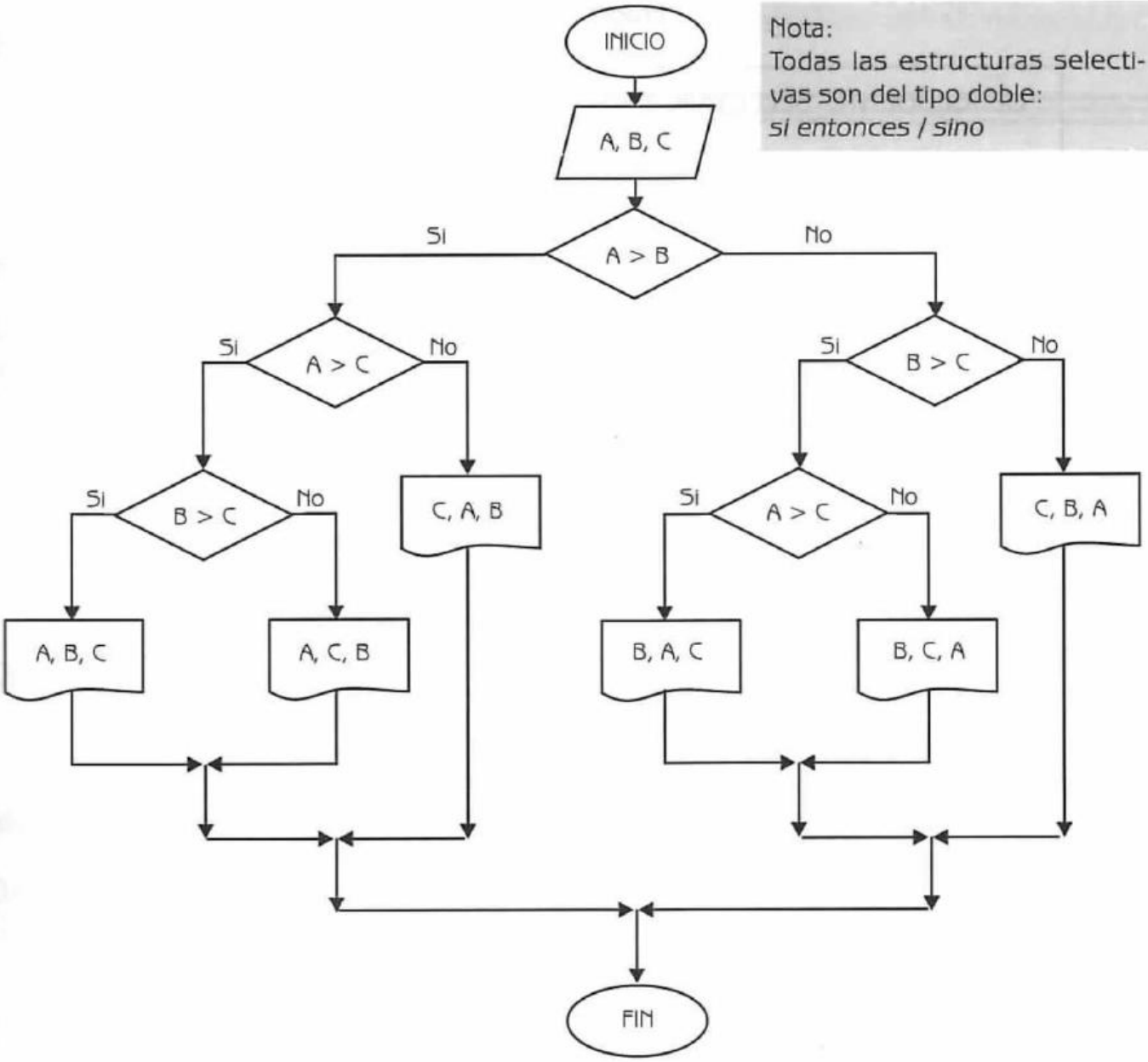
A continuación en el ejemplo 2.12, presentamos un problema donde el lector puede aplicar los conceptos estudiados en estructuras selectivas en cascada.

Ejemplo 2.12

Dados los datos A, B y C que representan números enteros diferentes, construya un diagrama de flujo para escribir estos números en forma descendente.

Datos: A, B, C (variables de tipo entero).

www.FreeLibros.com



Nota:
Todas las estructuras selectivas son del tipo doble:
si entonces / sino

Diagrama de Flujo 2.15

Explicación de las variables

A, *B* y *C*: Variables de tipo entero.

A continuación presentamos el diagrama de flujo en lenguaje algorítmico.

www.FreeLibros.com

Programa 2.15

```
DESCENDENTE_SELECTIVA_CASCADA

{El programa, dados como datos tres números enteros diferentes,
escribe estos números en forma descendente.

{A, B y C son variables de tipo entero}

1. Leer A, B, C
2. Si A > B
        entonces
        2.1.  Si A > C
                entonces
                2.1.1.  Si B > C
                                entonces
                                    Escribir A, B y C
                                sino
                                    Escribir A, C y B
                2.1.2.  {Fin del condicional del paso 2.1.1}
                sino
                    Escribir C, A y B
        2.2.  {Fin del condicional del paso 2.1}
        sino
        2.3.  Si B > C
                entonces
                2.3.1.  Si A > C
                                entonces
                                    Escribir B, A y C
                                sino
                                    Escribir B, C y A
                2.3.2.  {Fin del condicional del paso 2.3.1}
                sino
                    Escribir C, B y A
      2.4.  {Fin del condicional del paso 2.3}
3. {Fin del condicional del paso 2}
```

www.FreeLibros.com

Problemas resueltos

Problema 2.1

El número de sonidos emitidos por un grillo en un minuto, es una función de la temperatura. Como resultado de esto, es posible determinar el nivel de la temperatura haciendo uso de un grillito como termómetro.

La fórmula para la función es:

$$T = N / 4 + 40$$

Donde: T representa la temperatura en grados Fahrenheit y N el número de sonidos emitidos por minuto

Construya un diagrama de flujo que le permita calcular la temperatura, teniendo en cuenta el número de sonidos emitidos por el grillo.

Dato: N

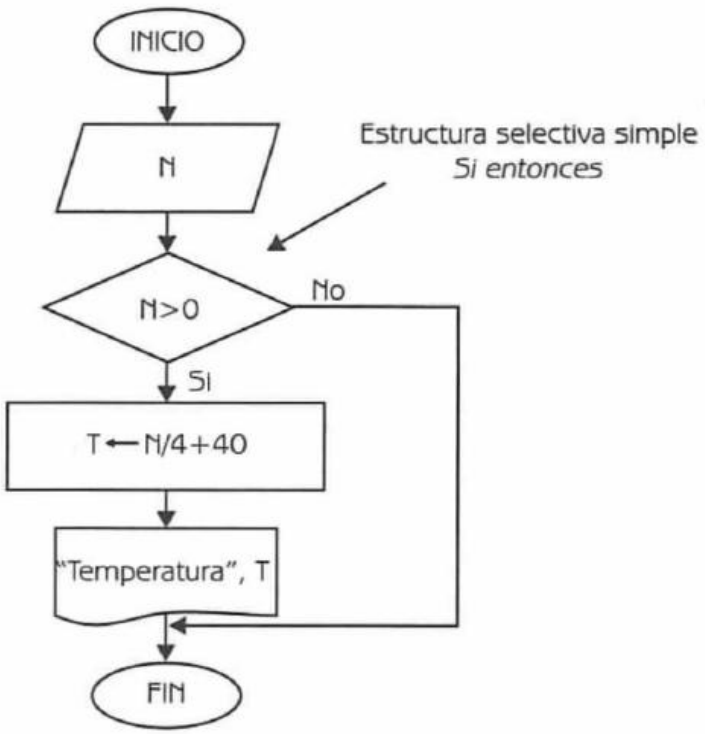


Diagrama de Flujo 2.16

www.FreeLibros.com

Explicación de las variables

N: Variable de tipo entero.

T: Variable de tipo real. Almacena la temperatura en grados Fahrenheit.

A continuación en la tabla 2.8 observamos el seguimiento del algoritmo para diferentes corridas.

Tabla 2.8

NUMERO DE CORRIDA	DATO	RESULTADO
	N	T
1	8	42.00
2	15	43.75
3	11	42.75
4	25	46.25
5	–5	
6	50	52.50

: Expresa valores que se imprimen.

Programa 2.16

```
GRILLO_TEMPERATURA

{El programa, dado como dato el número de sonidos emitidos por un
grillo en un minuto, calcula la temperatura en grados Fahrenheit}

{N es una variable de tipo entera. T es una variable de tipo real}

1. Leer N
2. Si N > 0 entonces
        Hacer T ⟵ N/4 + 40
        Escribir "Temperatura", T
3. {Fin del condicional del paso 2}
```

Problema 2.2

Construya un diagrama de flujo tal que dado como datos los valores enteros P y Q, determine si los mismos satisfacen la siguiente expresión:

www.FreeLibros.com

$$P^3 + Q^4 - 2*P^2 < 680$$

En caso afirmativo debe imprimir los valores P y Q.

Datos: P,Q (variables de tipo entero que expresan los datos que se ingresan).

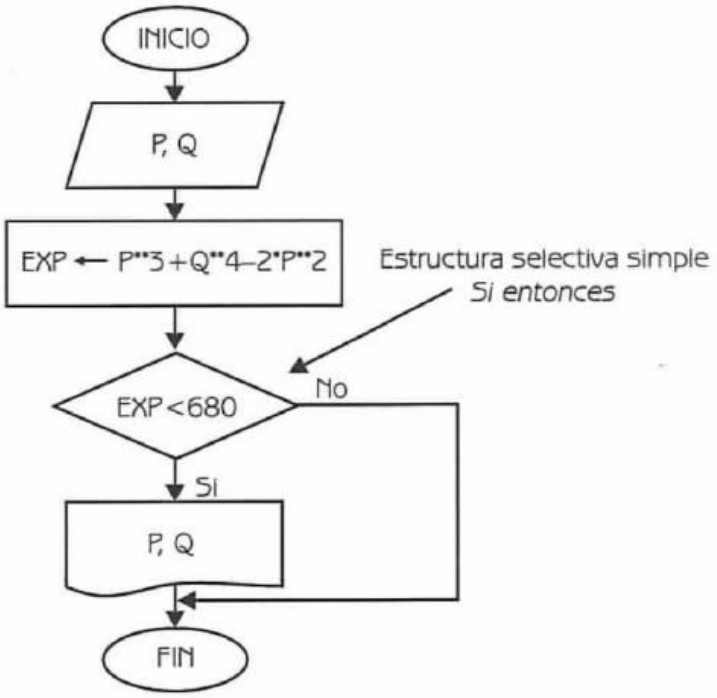


Diagrama de Flujo 2.17

Explicación de las variables

P, Q: Variables de tipo entero.
EXP: Variable de tipo real. Almacena el resultado del cálculo de la expresión.

A continuación en la tabla 2.9 podemos observar el seguimiento del algoritmo.

Tabla 2.9

NUMERO DE CORRIDA	DATOS		CALCULO AUXILIAR	RESULTADO	
	P	Q		P	Q
1	3	5	634	3	5
2	6	8	4240		
3	2	4	256	2	4
4	7	5	870		
5	2	6	1296		

: Expresa valores que se imprimen.

www.FreeLibros.com

Programa 2.17

```
EXPRESION

{El programa dado como datos dos valores enteros, determina si los
mismos satisfacen una expresión}

{ P y Q son variables de tipo entero. EXP es una variable de tipo real}

1. Leer P, Q
2. Hacer EXP ⟵ P**3 + Q**4 – 2 * P**2
3. Si EXP < 680 entonces
        Escribir P, Q
4. {Fin del condicional del paso 3}
```

Problema 2.3

Las raíces reales de la expresión $ax^2 + bx + c = 0$ se obtienen a través de la fórmula:

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Fórmula 2.2

Haga el diagrama de flujo para calcular las raíces reales, de ser posible, de una ecuación de segundo grado.

Datos: A, B, C con $A \neq 0$

Donde:

A, B y C son variables de tipo real. Representan los coeficientes de la ecuación.

www.FreeLibros.com

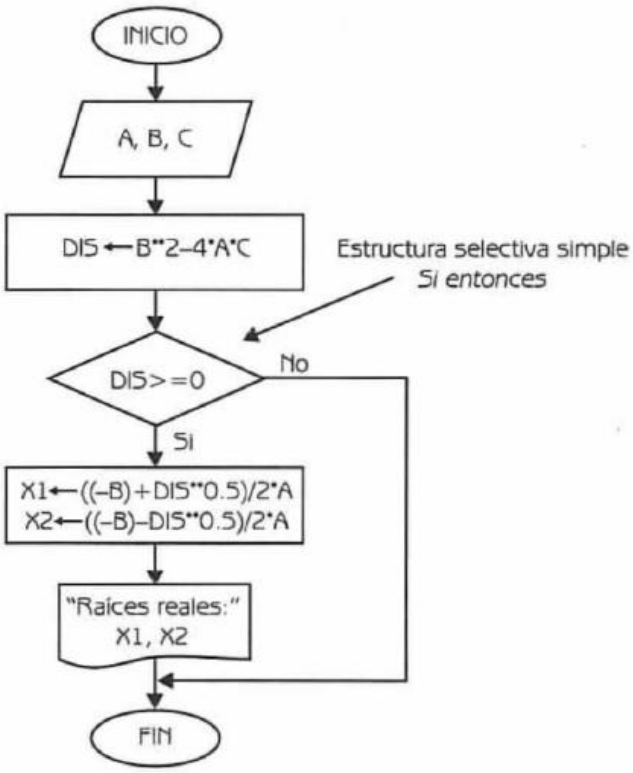


Diagrama de Flujo 2.18

Explicación de las variables

A, B, C: Variables de tipo real.

DIS: Variable de tipo real. Almacena el discriminante de la ecuación.

X1: Variable de tipo real. Almacena la primera raíz real de la ecuación.

X2: Variable de tipo real. Almacena la segunda raíz real de la ecuación.

A continuación en la tabla 2.10 mostramos el seguimiento del algoritmo para diferentes corridas.

Tabla 2.10

NUMERO DE CORRIDA	DATOS			CALCULO AUXILIAR	RESULTADOS	
	A	B	C	DIS	X1	X2
1	3	9	1.5	63	-0.17	-2.82
2	8	4	5	-144		
3	2.5	-6	-4	76	2.94	-0.54
4	-7.5	3	1	39	-0.21	0.61
5	-4	5	-3	-23		

: Expresa valores que se imprimen.

www.FreeLibros.com

Programa 2.18

```
ECUACION_SEGUNDO_ORDEN

{El programa, dado como datos los coeficientes de la ecuación,
calcula las raíces reales – si existen–}

{A, B, C, DIS, X1 y X2 son variables de tipo real}

1. Leer A, B y C
2. Hacer DIS ←— B**2 – 4 *A* C
3. Si DIS > 0 entonces
       Hacer X1 ←— ((–B)+ DIS**0.5)/2*A y
            X2 ←— ((–B) – DIS**0.5)/2*A
       Escribir "Raíces reales", X1, X2
4. {Fin del condicional del paso 3}
```

Problema 2.4

Construya un diagrama de flujo tal que dado como datos la matrícula y 5 calificaciones de un alumno; imprima la matrícula, el promedio y la palabra "aprobado" si el alumno tiene un promedio mayor o igual que 6, y la palabra "no aprobado" en caso contrario.

Datos: MAT, CAL1, CAL2, CAL3, CAL4, CAL5

Donde:

MAT	es una variable entera que representa la matrícula del alumno.
CAL1, CAL2, CAL3, CAL4 y CAL5	son variables de tipo real que representan las 5 calificaciones del alumno.

www.FreeLibros.com

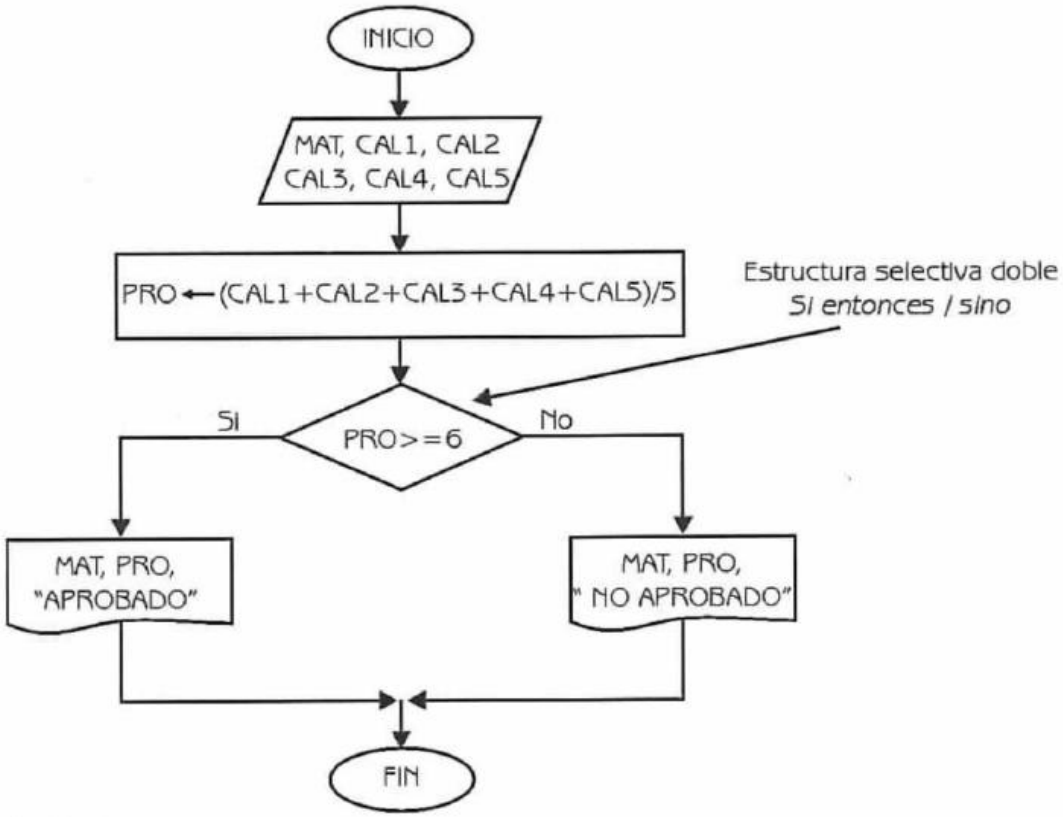


Diagrama de Flujo 2.19

Explicación de las variables

MAT: Variable de tipo entero.

CAL1, CAL2, CAL3, CAL4 y CAL5: Variables de tipo real.

PRO: Variable de tipo real. Almacena el promedio de las 5 calificaciones.

A continuación en la siguiente tabla, se puede observar el seguimiento del algoritmo para diferentes corridas.

www.FreeLibros.com

Tabla 2.11

NUMERO DE CORRIDA	DATOS						RESULTADOS	
	MAT	CAL1	CAL2	CAL3	CAL4	CAL5	PRO	COMENTARIO
1	16500	6	7.50	8	9.50	7	7.60	"APROBADO"
2	16650	5	4.80	7	6.30	5.90	5.80	"NO APROBADO"
3	17220	8.60	9	9	5.9	6.30	7.76	"APROBADO"
4	18240	7	4.60	4.90	7	5.60	5.82	"NO APROBADO"
5	17246	8	8.50	8.30	9.20	9.30	8.66	"APROBADO"
6	18250	9	9.25	8.10	9.80	10	9.23	"APROBADO"

: Expresa valores que se imprimen

Programa 2.19

```
PROMEDIO_ALUMNO

{El programa, dado como datos la matrícula y calificaciones de un alumno; imprime la matrícula, el promedio y "aprobado" o "no aprobado", dependiendo si su promedio fue mayor o igual que 6 o menor que 6, respectivamente}

{MAT es una variable de tipo entero. CAL1, CAL2, CAL3, CAL4, CAL5 y PRO son variables de tipo real}

1. Leer MAT, CAL1, CAL2, CAL3, CAL4 y CAL5
2. Hacer PRO ⟵ (CAL1 + CAL2 + CAL3 + CAL4 + CAL5) / 5
3. Si PRO > 6
        entonces
            Escribir MAT, PRO, "APROBADO"
        sino
            Escribir MAT, PRO, "NO APROBADO"
4. {Fin del condicional del paso 3}
```

Problema 2.5

Construya un diagrama de flujo tal que dado como dato un número entero, determine e imprima si el mismo es positivo, negativo o nulo.

Dato: NUM (variable entera que representa el número que se ingresa).

www.FreeLibros.com

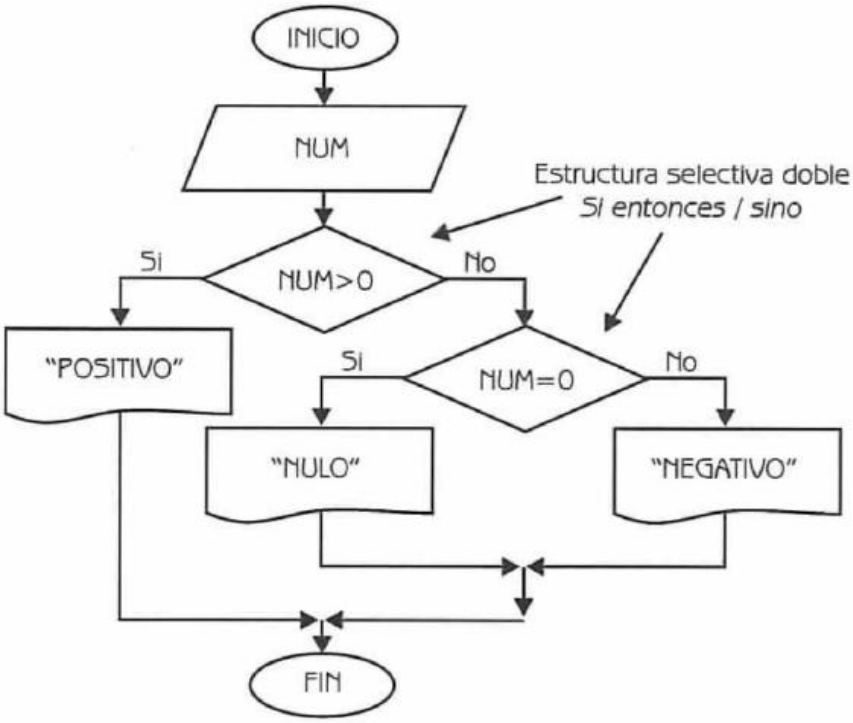


Diagrama de Flujo 2.20

Explicación de las variables

NUM: Variable de tipo entera.

Tabla 2.12

NUMERO DE CORRIDA	DATO	RESULTADO
	NUM	
1	5	"Positivo"
2	2	"Positivo"
3	0	"Nulo"
4	–7	"Negativo"
5	–15	"Negativo"

: Expresa valores que se imprimen

Programa 2.20

```
POSITIVO_NEGATIVO_NULO

{El programa dado como dato un número entero, determina si el
mismo es positivo, negativo o nulo}
```

www.FreeLibros.com

```
{NUM es una variable de tipo entero.}

1. Leer NUM
2. Si NUM > 0
        entonces
            Escribir "Positivo"
        sino
        2.1 Si NUM = 0
            entonces
                Escribir "Nulo"
            sino
                Escribir "Negativo"
        2.2 {Fin del condicional del paso 2.1}
3. {Fin del condicional del paso 2}
```

Problema 2.6

Dado un número entero A, haga un diagrama de flujo para determinar si el mismo es par, impar o nulo.

Dato: A (variable de tipo entero).

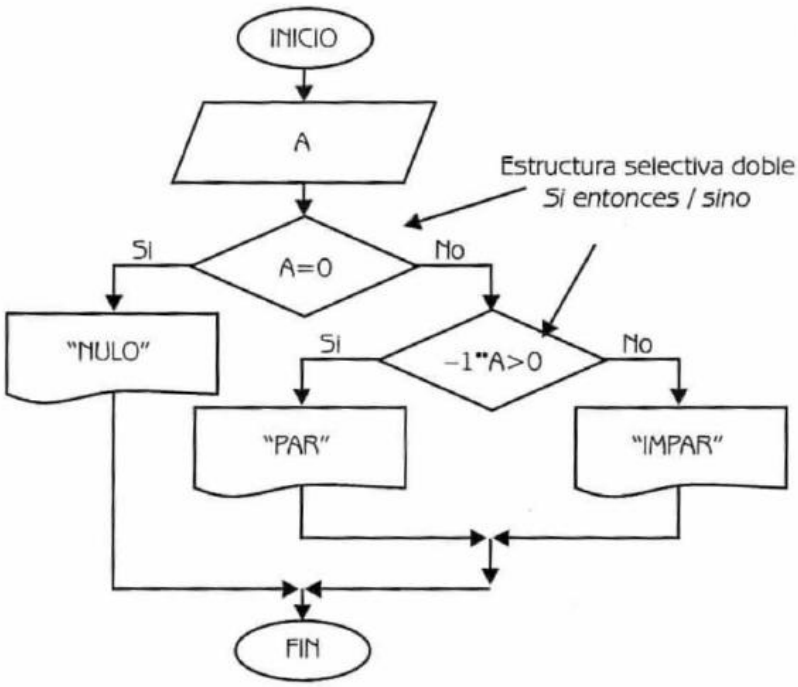


Diagrama de Flujo 2.21

Nota:
Utilizamos la expresión (–1** A) para establecer si el número es par o impar. –1 elevado a un número par, da un valor positivo. –1 elevado a un número impar, da un valor negativo. La condición para establecer si el número es nulo, debe ir al principio, porque de otra forma si ingresamos el número 0 (cero), daría como resultado al aplicar la expresión: par. Recordemos que todo número elevado a la cero, da 1.

www.FreeLibros.com

Explicación de las variables

A: Variable de tipo entero.

Programa 2.21

```
PAR_IMPAR_NULO

{El programa dado como dato un número entero, determina si el
mismo es par, impar o nulo}

{A es una variable de tipo entero}

1. Leer A
2. Si A = 0
       entonces
            Escribir "Nulo"
       sino
       2.1 Si (-1 ** A) > 0
                 entonces
                      Escribir "Par"
                 sino
                      Escribir "Impar"
       2.2 {Fin del condicional del paso 2.1}
3. {Fin del condicional del paso 2}
```

Problema 2.7

Construya un diagrama de flujo tal que dado como datos de entrada tres números enteros, determine si los mismos están en orden creciente.

Datos: A, B, C (variables de tipo entero. Los números son diferentes entre sí).

www.FreeLibros.com

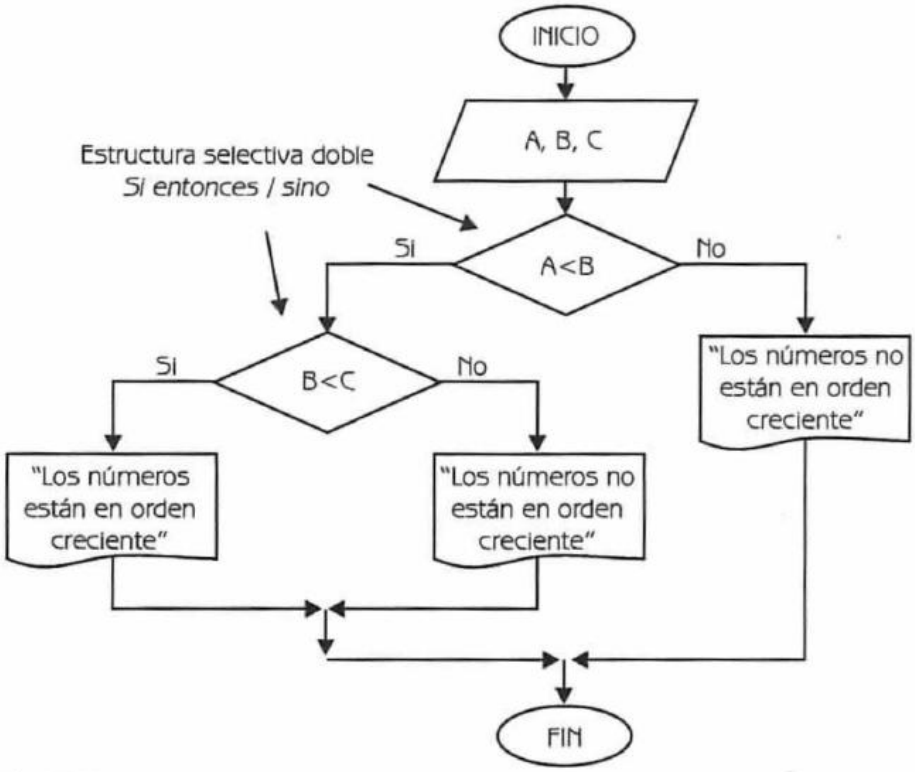


Diagrama de Flujo 2.22

Explicación de las variables.

A, *B* y *C*: Variables de tipo entero.

A continuación presentamos el programa correspondiente.

www.FreeLibros.com

Programa 2.22

```
ORDEN_CRECIENTE

{El programa dados como datos tres números enteros, determina si
los mismos están en orden creciente}

 {A, B y C son variables de tipo entero}

1. Leer A, B y C
2. Si A < B
     entonces
     2.1 Si B < C
          entonces
               Escribir "Los números están en orden creciente"
          sino
               Escribir "Los números no están en orden creciente"
     2.2 {Fin del condicional del paso 2.1}
     sino
               Escribir "Los números no están en orden creciente"
3. {Fin del ciclo del paso 2}
```

Problema 2.8

En una tienda efectúan un descuento a los clientes dependiendo del monto de la compra. El descuento se efectúa con base en el siguiente criterio:

Si el monto es menor que \$500 ⟶ no hay descuento.

Si el monto está comprendido entre \$500 y \$1 000 inclusive ⟶ 5% de descuento.

Si el monto está comprendido entre \$1 000 y \$7 000 inclusive ⟶ 11% de descuento.

Si el monto está comprendido entre \$7 000 y \$15 000 inclusive ⟶ 18% de descuento.

Si el monto es mayor a \$15 000 ⟶ 25% de descuento.

Construya un diagrama de flujo tal que dado el monto de la compra de un cliente, determine lo que el mismo debe pagar.

Dato: COMPRA (variable de tipo real que representa el monto de la compra).

www.FreeLibros.com

Nota:
Todas las estructuras selectivas son del tipo doble:
Si entonces / sino

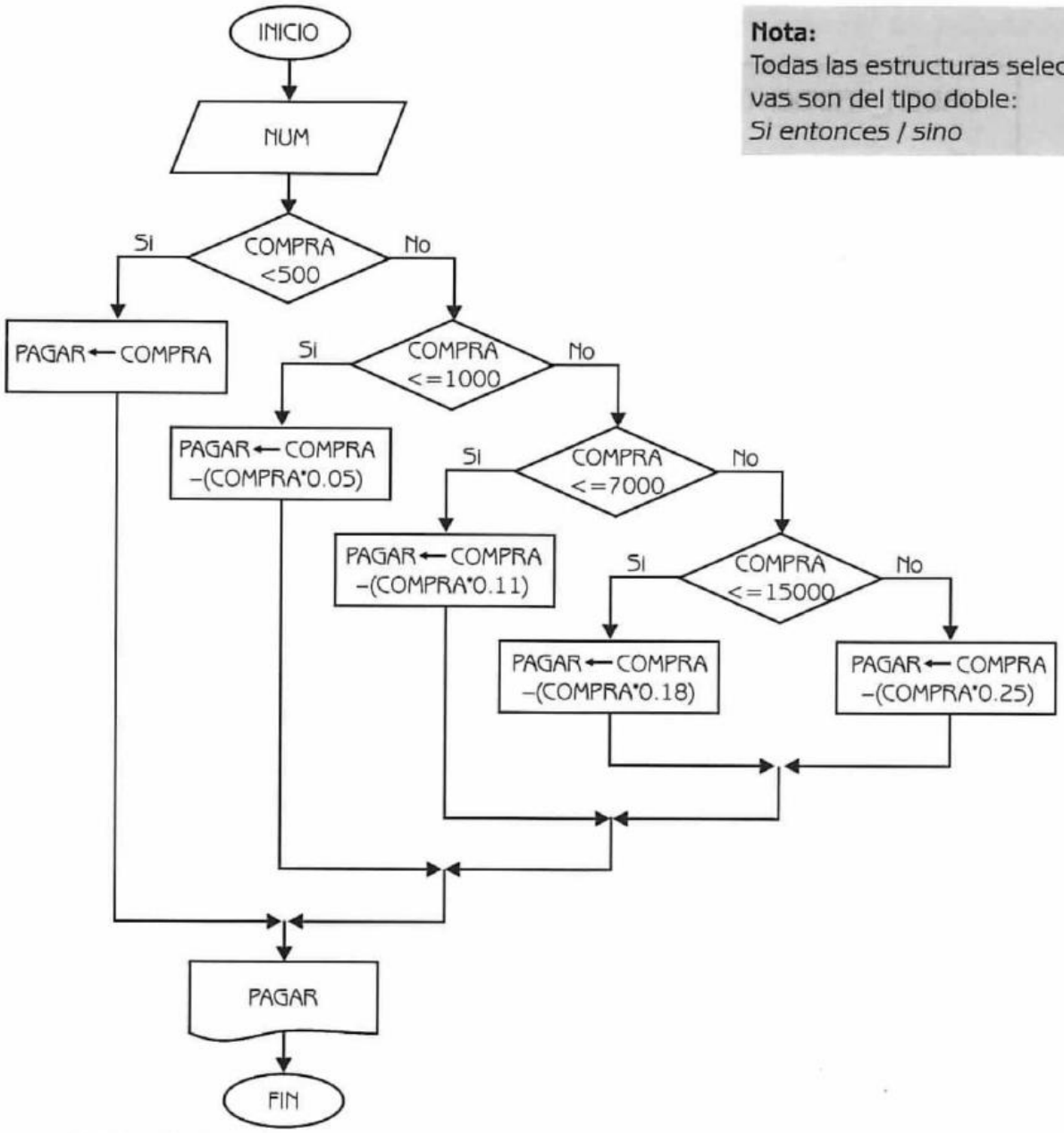


Diagrama de Flujo 2.23

Explicación de las variables

COMPRA: Variable de tipo real.

PAGAR: Variable de tipo real. Expresa lo que debe pagar el cliente, teniendo en cuenta los descuentos correspondientes.

A continuación en la tabla 2.13 presentamos el seguimiento del diagrama de flujo, para diferentes corridas

www.FreeLibros.com

Tabla 2.13		
NUMERO DE CORRIDA	DATO	RESULTADO
	COMPRA	PAGAR
1	3500.00	3115.00
2	6850.00	6096.50
3	375.80	375.80
4	690.50	655.97
5	12350.00	10127.00
6	25314.18	18985.63
7	3750.00	3337.50
8	14200.50	11644.41
9	895.80	851.01
10	1318.50	1173.46

: Expresa valores que se imprimen.

Programa 2.23

```
TIENDA_DESCUENTOS

{El programa dado como dato el monto de la compra de un cliente, determina lo
que el mismo debe pagar teniendo en cuenta una serie de descuentos}

{COMPRA y PAGAR son variables del tipo real}

1. Lee COMPRA
2. Si COMPRA < 500
       entonces
           Hacer PAGAR ←— COMPRA
       sino
    2.1 Si COMPRA <= 1000
           entonces
               Hacer PAGAR ←— COMPRA – ( COMPRA * 0.05)
           sino
        2.1.1 Si COMPRA <= 7000
                 entonces
                     Hacer PAGAR ←— COMPRA – ( COMPRA * 0.11)
                 sino
           2.1.1.1 Si COMPRA <= 15000
                       entonces
                           Hacer PAGAR ←— COMPRA–( COMPRA*0.18)
                       sino
                           Hacer PAGAR ←— COMPRA–( COMPRA*0.25)
           2.1.1.2 {Fin del condicional del paso 2.1.1.1}
        2.1.2 {Fin del condicional del paso 2.1.1}
```

www.FreeLibros.com

```
    2.2 {Fin del condicional del paso 2.1}
3. {Fin del condicional del paso 2}
4. Escribir PAGAR
```

Problema 2.9

En un cierto país el impuesto que se debe pagar por los artículos se calcula mediante la siguiente regla: los primeros $20 no causan impuesto, los siguientes $20 tienen el 30% de impuesto y el resto el 40% de impuesto, pero si el costo del producto es mayor a $500, entonces en lugar del 40% se cobra el 50%.

Diseñe un diagrama de flujo que lea el costo básico de un artículo y calcule su precio total (precio total = precio básico + impuesto).

Dato: PREBAS (variable de tipo real que representa el precio básico del producto que se ingresa).

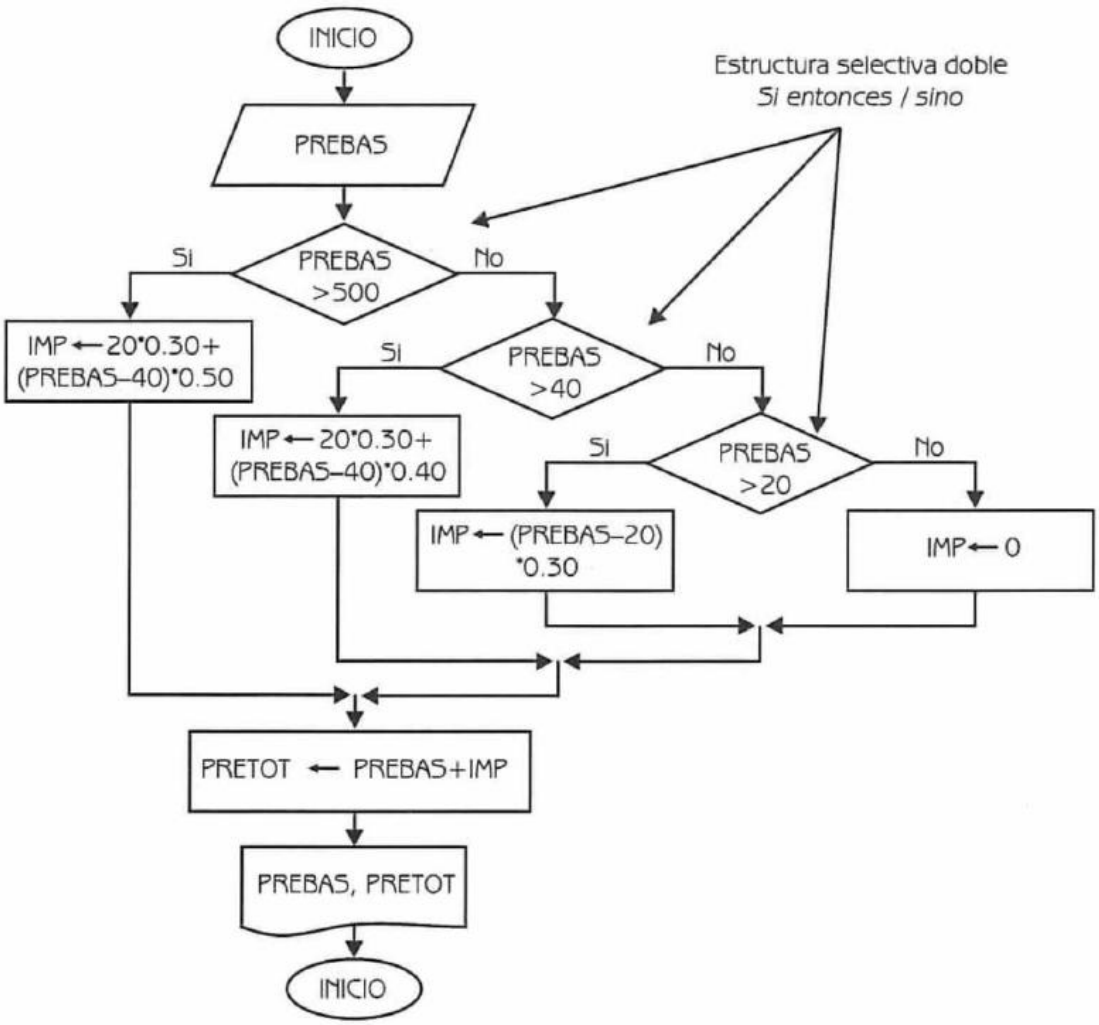


Diagrama de Flujo 2.24

www.FreeLibros.com

Explicación de las variables

PREBAS: Variable de tipo real.
IMP: Variable de tipo real. Almacena el impuesto del producto que se ingresa.
PRETOT: Variable de tipo real. Almacena el costo total del producto, es decir el costo básico más el impuesto.

A continuación en la siguiente tabla podemos observar el seguimiento del algoritmo para diferentes corridas.

Tabla 2.14

NUMERO DE CORRIDA	PREBAS	IMP	PRETOT
1	75.68	20.27	95.95
2	17.33	0	17.33
3	128.40	41.36	169.76
4	27.90	2.37	30.27
5	527.50	249.75	777.25
6	217.80	77.12	294.92

: Expresa valores que se imprimen.

```
IMPUESTO_ARTICULO

{El programa dado como dato el precio básico de un artículo, calcula el
impuesto correspondiente del mismo teniendo en cuenta ciertos criterios}

{PRUEBAS, IMP y PRETOT son variables de tipo real}

1. Leer PREBAS
2. Si  PREBAS > 500
        entonces
                Hacer IMP ⟵ 20 * 0.30 + (PREBAS – 40) * 0.50
        sino
            2.1 Si PREBAS > 40
                  entonces
                     Hacer IMP ⟵ 20 * 0.30 + (PREBAS – 40) * 0.40
                  sino
                2.1.1 Si PREBAS > 20
                        entonces
                           Hacer IMP ⟵ (PREBAS – 20) * 0.30
                        sino
```

www.FreeLibros.com

```
                    Hacer IMP ⟵ 0
            2.1.2 {Fin del condicional del paso 2.1.1}
        2.2 {Fin del condicional del paso 2.1}
3. {Fin del condicional del paso 2}
4. Hacer PRETOT ⟵ PREBAS + IMP
5. Escribir PREBAS, PRETOT
```

Problema 2.10

Dado como datos tres números reales, identifique cuál es el mayor. Considere que los números pueden ser iguales. Desarrolle el diagrama de flujo correspondiente.

Datos: A, B, C (variables de tipo real).

Nota:
Todas las estructuras selectivas son del tipo doble: *Si entonces / sino*.

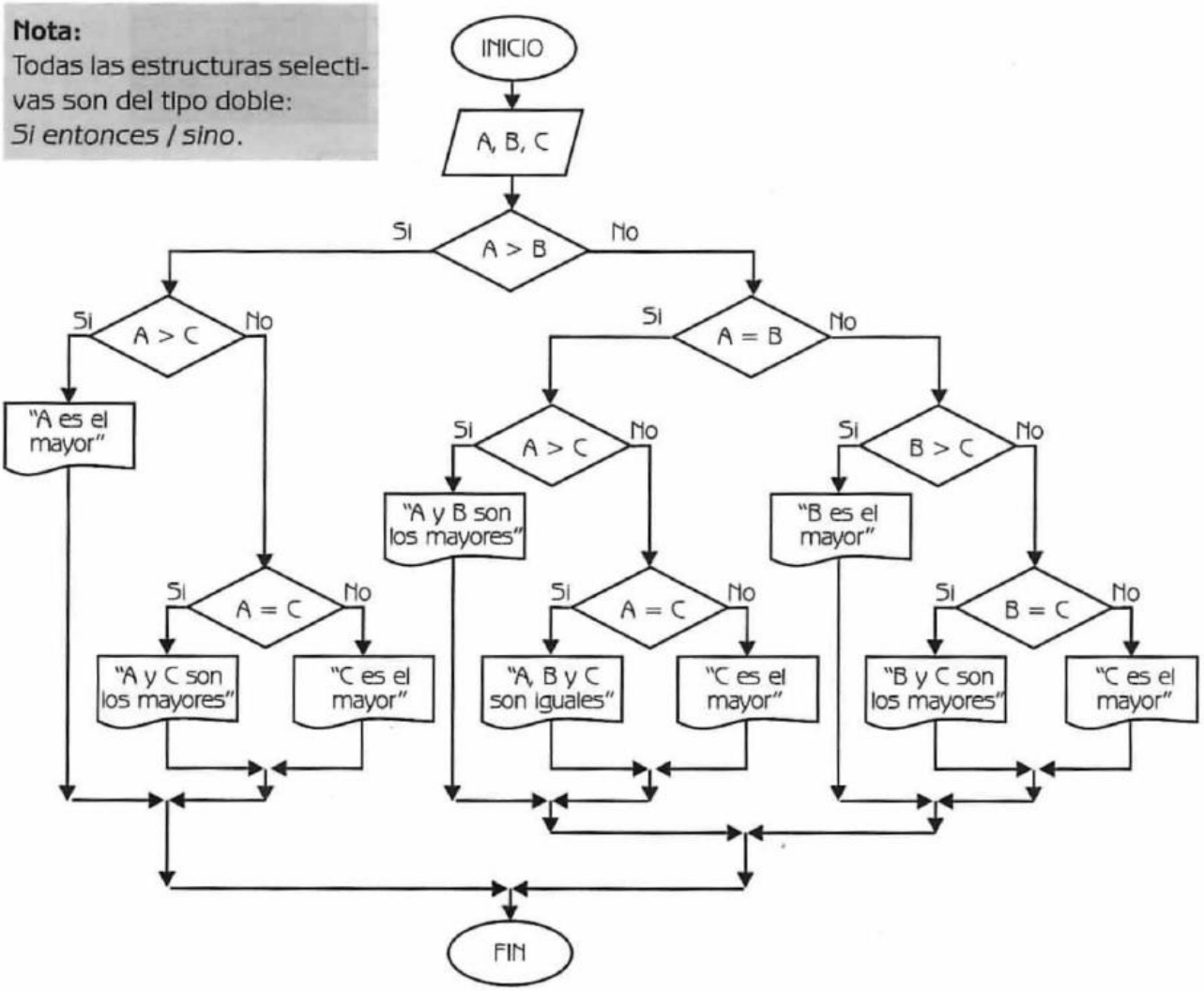


Diagrama de Flujo 2.25

www.FreeLibros.com

Explicación de las variables

A, *B* y *C*: Variables de tipo entero.

Tabla 2.15

NUMERO DE CORRIDA	DATOS			RESULTADO
	A	B	C	
1	6	8	9	"C es el mayor"
2	7	7	4	"A y B son los mayores"
3	15	22	15	"B es el mayor"
4	48	45	45	"A es el mayor"
5	17	17	17	"A, B y C son iguales"

: Expresa valores que se imprimen.

A continuación presentamos el diagrama de flujo en lenguaje algorítmico:

```
MAYOR_TRES_NUMEROS_COMPLEJO

{El programa, dados como datos tres números cuyos valores pueden ser iguales, determina cuál es el mayor o los mayores}

{A, B y C son variables de tipo real}

1. Leer A, B, C
2. Si A > B
   entonces
   2.1 Si A > C
         entonces
            Escribir "A es el mayor"
         sino
      2.1.1 Si A=C
               entonces
                  Escribir "A y C son los mayores"
               sino
                  Escribir "C es el mayor"
      2.1.2 {Fin del condicional del paso 2.1.1}
   2.2 {Fin del condicional del paso 2.1}
   sino
   2.3 Si A = B
```

www.FreeLibros.com

```
        entonces
        2.3.1 Si A > C
                    entonces
                        Escribir "A y B son los mayores"
                    sino
                    2.3.1.1 Si A = C
                                entonces
                                    Escribir "A, B y C son los mayores"
                                sino
                                    Escribir "C es el mayor"
                    2.3.1.2 {Fin del condicional del paso 2.3.1.1}
        2.3.2 {Fin del condicional del paso 2.3.1}
        sino
        2.3.3 Si B > C
                    entonces
                        Escribir " B es el mayor"
                    sino
                    2.3.3.1 Si B = C
                                entonces
                                    Escribir "B y C son los mayores"
                                sino
                                    Escribir "C es el mayor"
                    2.3.3.2 {Fin del condicional del paso 2.3.3.1}
        2.3.4 {Fin del condicional del paso 2.3.3}
    2.4 {Fin del condicional del paso 2.3}
3. {Fin del condicional del paso 2}
```

Problema 2.11

El costo de las llamadas telefónicas internacionales depende de la zona geográfica en la que se encuentre el país destino y del número de minutos hablados. En la siguiente tabla se presenta el costo del minuto por zona. A cada zona se le ha asociado una clave.

Tabla 2.16

CLAVE	ZONA	PRECIO
12	América del Norte	2
15	América Central	2.2
18	América del Sur	4.5
19	Europa	3.5
23	Asia	6
25	África	6
29	Oceanía	5

www.FreeLibros.com

Construya un diagrama de flujo que le permita calcular e imprimir el costo total de una llamada.

Datos: CLAVE, NUMIN

Donde:

CLAVE es una variable entera que representa la clave de la zona geográfica a la que se llamó.

NUMIN es una variable entera que expresa la duración (en minutos) de la llamada.

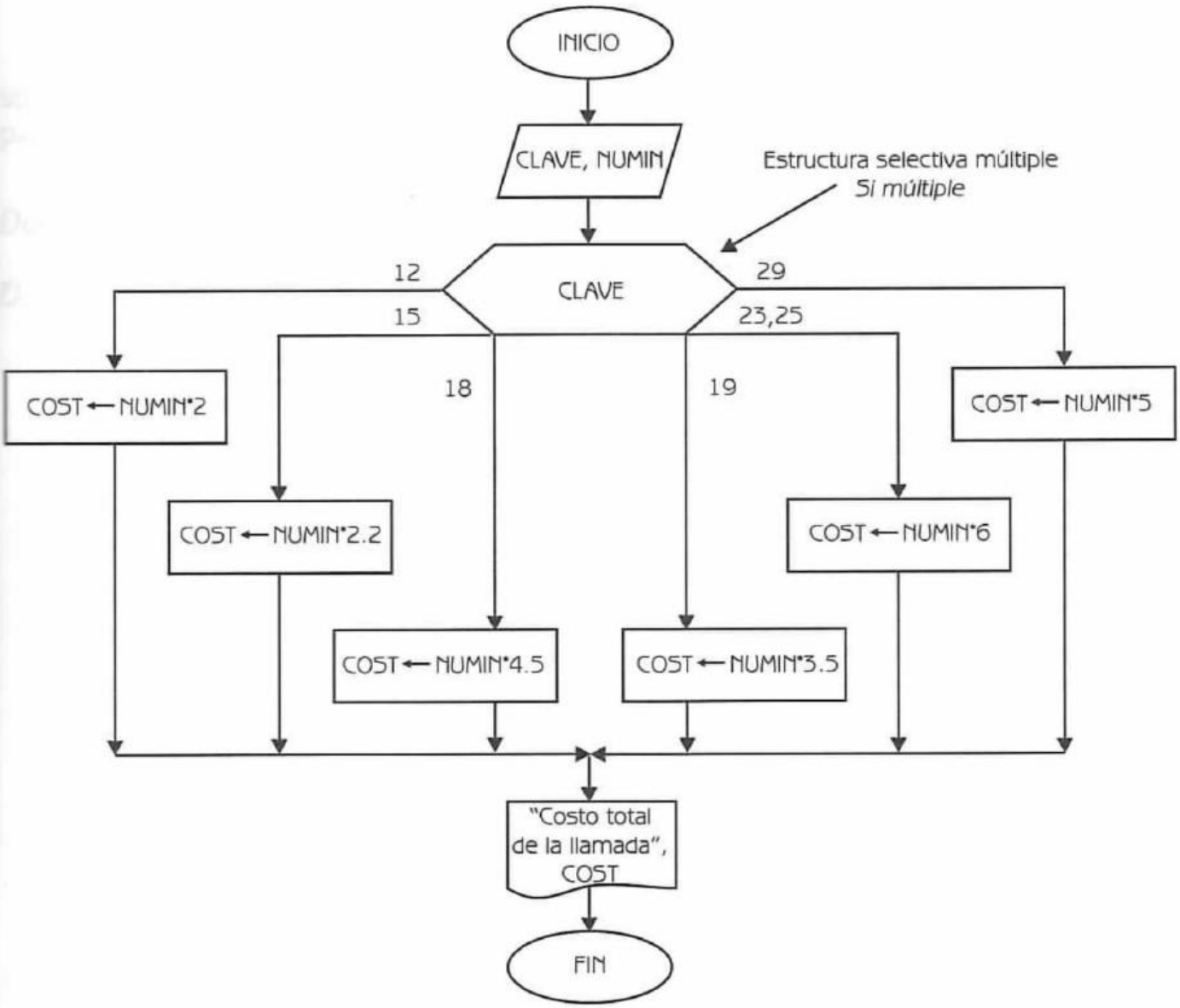


Diagrama de Flujo 2.26

www.FreeLibros.com

Explicación de las variables

CLAVE, NUMIN: Variables de tipo entero.
COST: Variable de tipo real. Almacena el costo total de la llamada telefónica.

En la tabla 2.17 podemos observar el seguimiento del algoritmo para diferentes corridas.

Tabla 2.17

NUMERO DE CORRIDA	DATOS		RESULTADO
	CLAVE	NUMIN	
1	23	5	30
2	15	4	8.8
3	15	10	22
4	29	8	40
5	18	12	54

: Expresa valores que se imprimen

Programa 2.26

```
LLAMADAS_TELEFONICAS

{El programa dados como dato la clave de la zona a la cual se llamó y el
número de minutos que duró la llamada, calcula el costo total de la mis-
ma}

{CLAVE y NUMIN son variables de tipo entero. COST es una variable
de tipo real}

1. Leer CLAVE y NUMIN
2. Si CLAVE igual
        12:   Hacer COST ←— NUMIN * 2
        15:   Hacer COST ←— NUMIN * 2.2
        18:   Hacer COST ←— NUMIN * 4.5
        19:   Hacer COST ←— NUMIN * 3.5
        23, 25:   Hacer COST ←— NUMIN * 6
        29:   Hacer COST ←— NUMIN * 5
3. {Fin del condicional del paso 2}
4. Escribir "Costo total de la llamada", COST
```

www.FreeLibros.com

Problema 2.12

Escriba un diagrama de flujo que permita calcular lo que hay que pagarle a un trabajador teniendo en cuenta su sueldo y las horas extras trabajadas. Para el pago de horas extras se toma en cuenta la categoría del trabajador.

Tabla 2.18

CATEGORIA	PRECIO HORA EXTRA
1	$ 30
2	$ 38
3	$ 50
4	$ 70

Cada trabajador puede tener como máximo 30 horas extras, si tienen más sólo se les pagarán 30. A los trabajadores con categoría mayor a 4 no debemos pagarle horas extras.

Datos: SUE, CATE, HE

Donde:

SUE es una variable real que representa el sueldo básico del trabajador.

CATE es una variable de tipo entero que indica la categoría del trabajador ($1 \leq CATE \leq 8$).

HE es una variable de tipo entero que representa las horas extras trabajadas por el trabajador.

www.FreeLibros.com

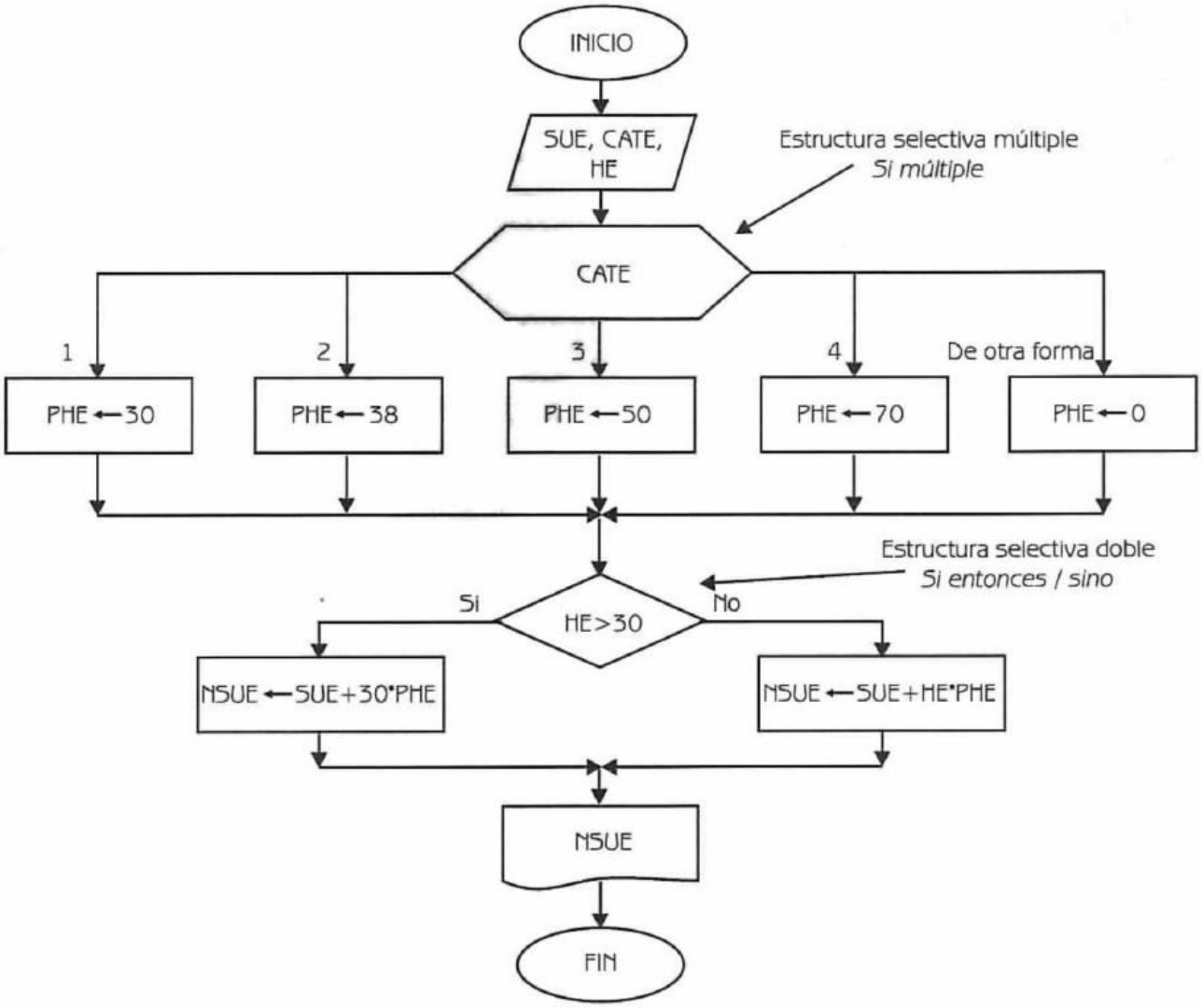


Diagrama de Flujo 2.27

Explicación de las variables

SUE: Variable de tipo real.

CATE: Variable de tipo entero.

HE: Variable de tipo entero.

PHE: Variable de tipo real. Almacena el costo de la hora extra, teniendo en cuenta la categoría del trabajador.

NSUE: Variable de tipo real. Almacena lo que hay que pagarle al trabajador teniendo en cuenta su sueldo y las horas extras trabajadas.

www.FreeLibros.com

Tabla 2.19

NUMERO DE CORRIDA	DATOS			RESULTADOS	
	SUE	CATE	HE	PHE	NSUE
1	1350	1	25	30	2100
2	2200	2	38	38	3340
3	980	1	15	30	1430
4	13200	8	22	0	13200
5	5800	5	17	0	5800
6	3350	3	49	50	4850
7	2480	2	14	38	3012
8	5125	4	3	70	5335
9	8700	6	21	0	8700
10	3200	3	8	50	3600

: Expresa valores que se imprimen

Programa 2.27

```
SUELDO_HORAS_EXTRAS

{El programa calcula lo que hay que pagarle a un trabajador teniendo en cuenta su
sueldo, horas extras y su categoría}

CATE y HE son variables de tipo entera. SUE, PHE y NSUE son variables de tipo real}

1. Leer SUE, CATE y HE
2. Si CATE igual
        1:  Hacer PHE ←— 30
        2:  Hacer PHE ←— 38
        3:  Hacer PHE ←— 50
        4:  Hacer PHE ←— 70
        De otra forma: Hacer PHE ←— 0
3. {Fin del condicional del paso 2}
4. Si HE > 30
        entonces
            Hacer NSUE ←— SUE + 30 * PHE
        sino
            Hacer NSUE ←— SUE + HE * PHE
5. {Fin del condicional del paso 4}
6. Escribir NSUE
```

Problema 2.13

Construya un diagrama de flujo tal que dado como datos la matrícula de un alumno, la carrera en la que está inscrito, su semestre y su promedio; determine

www.FreeLibros.com

si el mismo es apto para pertenecer a alguna de las facultades menores que tiene la universidad. Si el alumno es aceptado teniendo en cuenta las especificaciones que se listan abajo, se debe imprimir su matrícula, carrera y la palabra "aceptado".
Especificaciones para pertenecer a las facultades menores:

Economía:	Semestre ≥ 6 y promedio ≥ 8.8
Computación:	Semestre > 6 y promedio > 8.5
Administración:	Semestre > 5 y promedio > 8.5
Contabilidad:	Semestre > 5 y promedio > 8.5

Datos: MAT, CARR, SEM, PROM

Donde:

MAT	es una variable entera que representa la matrícula del alumno.
CARR	es una variable de tipo cadena de caracteres que expresa la carrera en la que está inscrito el alumno.
SEM	es una variable de tipo entero que representa el semestre que tiene aprobado el alumno.
PROM	es una variable de tipo real que expresa el promedio del alumno.

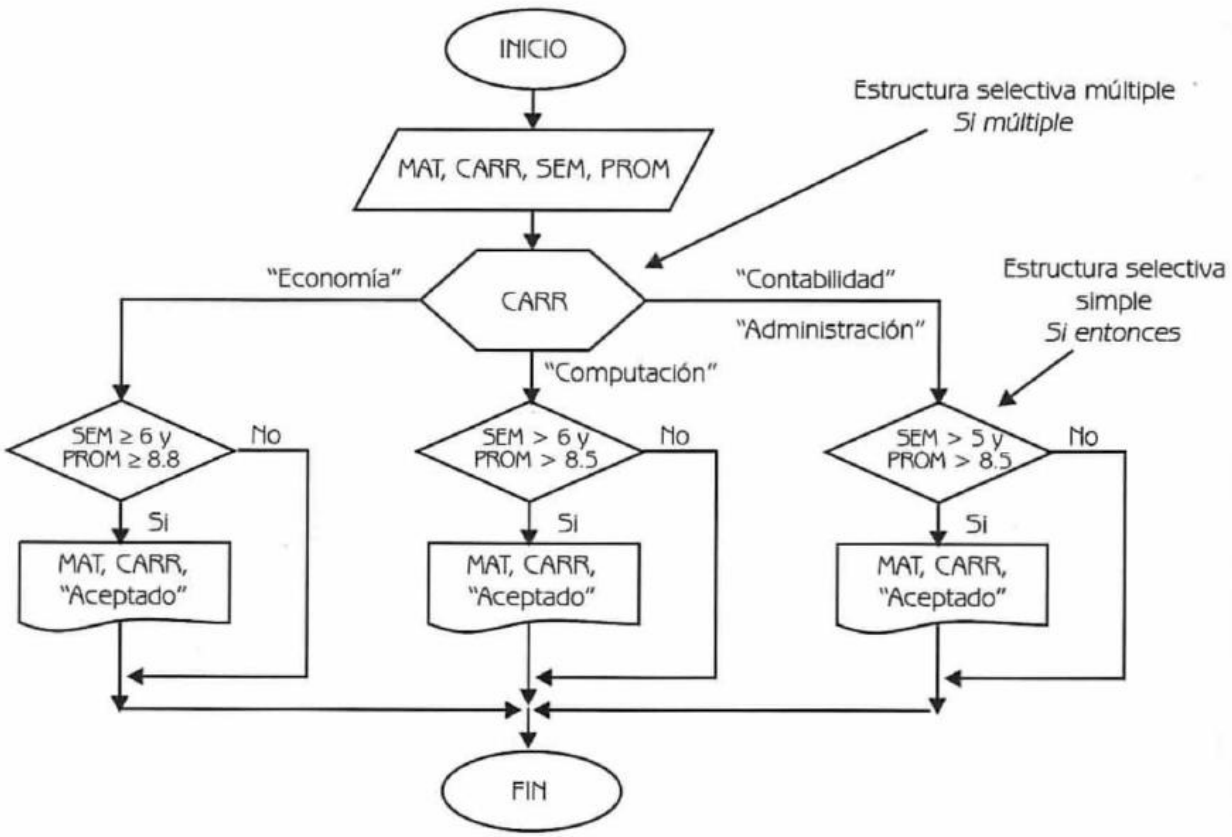


Diagrama de Flujo 2.28

www.FreeLibros.com

Explicación de las variables

MAT: Variable de tipo entera.
CARR: Variable de tipo cadena de caracteres.
SEM: Variable de tipo entera.
PROM: Variable de tipo real.

A continuación en la tabla 2.20 podemos observar el seguimiento del algoritmo para diferentes corridas.

Tabla 2.20

NUMERO DE CORRIDA	DATOS				RESULTADO
	MAT	CARR	SEM	PROM	
1	25900	"Contabilidad"	7	9	25900 Contabilidad Aceptado
2	25000	"Economía"	6	8.5	
3	25224	"Computación"	8	8.7	25224 Computación Aceptado
4	26000	"Administración"	6	9	26000 Administración Aceptado
5	26100	"Computación"	6	9.2	

: Expresa valores que se imprimen

Programa 2.28

```
FACULTAD_MENOR

{El programa dados como datos el número de la matrícula, carrera, semestre y
promedio de un alumno, determina si dicho alumno es aceptado o no como
miembro de la facultad menor}

{MAT y SUM son variables de tipo entero. PROM es una variable de tipo real.
CARR es una variable de tipo cadena de caracteres}

1. Leer MAT, CARR, SEM y PROM
2. Si CARR igual
      "Economía":      2.1 Si SEM ≥ 6 y PROM ≥ 8.8 entonces
                              Escribir MAT, CARR, "Aceptado"
                       2.2 {Fin del condicional del paso 2.1}
      "Computación":   2.3 Si SEM > 6 y PROM > 8.5 entonces
                              Escribir MAT, CARR, "Aceptado"
```

www.FreeLibros.com

```
                    2.4 {Fin del condicional del paso 2.3}
    "Administración",
    "Contabilidad":  2.5 Si SEM > 5 y PROM > 8.5 entonces
                            Escribir MAT, CARR, "Aceptado"
                    2.6 {Fin del condicional del paso 2.5}
3. {Fin del condicional del paso 2}
```

Problema 2.14

En un hospital se ha hecho un estudio sobre los pacientes registrados durante los últimos 10 años, con el objeto de hacer una aproximación de los costos de internación por paciente. Se obtuvo un costo promedio diario según el tipo de enfermedad que aqueja al paciente. Además se pudo determinar que en promedio todos los pacientes con edad entre 14 y 22 años implican un costo adicional del 10%. La siguiente tabla expresa los costos diarios, según el tipo de enfermedad.

Tabla 2.21

TIPO DE ENFERMEDAD	COSTO/PACIENTE/DIA
1	25
2	16
3	20
4	32

Construya un diagrama de flujo que calcule e imprima el costo total que representa un paciente.

Datos: TIPOENF, EDAD, DIAS

Donde:

TIPOENF es una variable entera representa el tipo de enfermedad del paciente.

EDAD es una variable entera que indica la edad del paciente.

DIAS es una variable entera que indica el número de días que el paciente estuvo hospitalizado.

www.FreeLibros.com

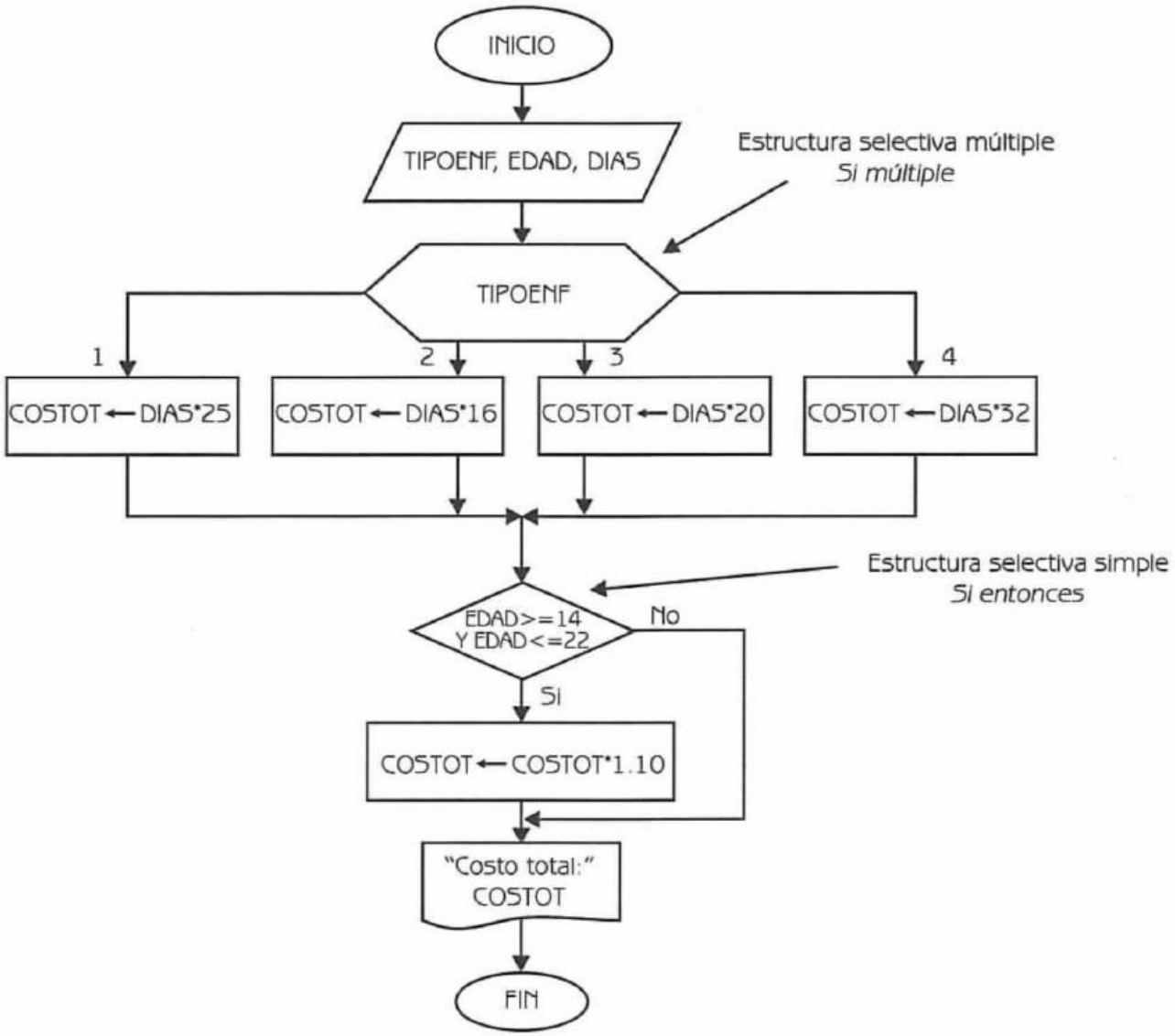


Diagrama de Flujo 2.29

Explicación de las variables

TIPOENF, EDAD, DIAS: Variables de tipo entero.

COSTOT: Variable de tipo real. Almacena el costo total por paciente.

A continuación en la tabla 2.22 podemos observar el seguimiento del algoritmo para diferentes corridas.

www.FreeLibros.com

Tabla 2.22

NUMERO DE CORRIDA	DATOS			RESULTADO
	TIPOENF	EDAD	DIAS	COSTOT
1	3	35	4	80
2	2	20	6	96 105.6
3	3	40	5	100
4	4	16	8	256 281.6

: Expresa valores que se imprimen

Programa 2.29

```
HOSPITAL

{El programa dados como datos la edad del paciente, el tipo de enfermedad
padecida y el número de días hospitalizados, calcula el costo total por
internación}

{TIPOENF, EDAD y DIAS son variables de tipo entero. COSTOT es una variable de
tipo real}

1. Leer TIPOENF, EDAD y DIAS
2. Si TIPOENF igual
       1: Hacer COSTOT ←— DIAS * 25
       2: Hacer COSTOT ←— DIAS * 16
       3: Hacer COSTOT ←— DIAS * 20
       4: Hacer COSTOT ←— DIAS * 32
3. {Fin del condicional del paso 2}
4. Si EDAD ≥ 14 y EDAD ≤ 22 entonces
       Hacer COSTOT ←— COSTOT * 1.10
5. {Fin del condicional del paso 4}
6. Escribir "Costo Total:", COSTOT
```

www.FreeLibros.com

3

Estructuras algorítmicas repetitivas

3.1 Introducción

Es muy común encontrar en la práctica algoritmos cuyas operaciones se deben ejecutar un número repetido de veces. Si bien las instrucciones son las mismas, los datos sobre los que se opera varían. El conjunto de instrucciones que se ejecuta repetidamente se llama ciclo.

Todo ciclo debe terminar de ejecutarse luego de un número finito de veces, por lo que es necesario en cada iteración del mismo, evaluar las condiciones necesarias para decidir si se debe seguir ejecutando o si debe detenerse. En todo ciclo, siempre debe existir una condición de parada o fin de ciclo.

En algunos algoritmos podemos establecer a priori que el ciclo se repetirá un número definido de veces. Es decir, el número de repeticiones no dependerá de las proposiciones dentro del ciclo. Llamaremos *repetir* a la estructura algorítmica repetitiva que se ejecuta un número definido de veces.

Por otra parte, en algunos algoritmos no podemos establecer a priori el número de veces que ha de ejecutarse el ciclo, sino que este número dependerá de las proposiciones dentro del mismo. Llamaremos *mientras* a la estructura algorítmica repetitiva que se ejecuta mientras la condición evaluada resulta verdadera.

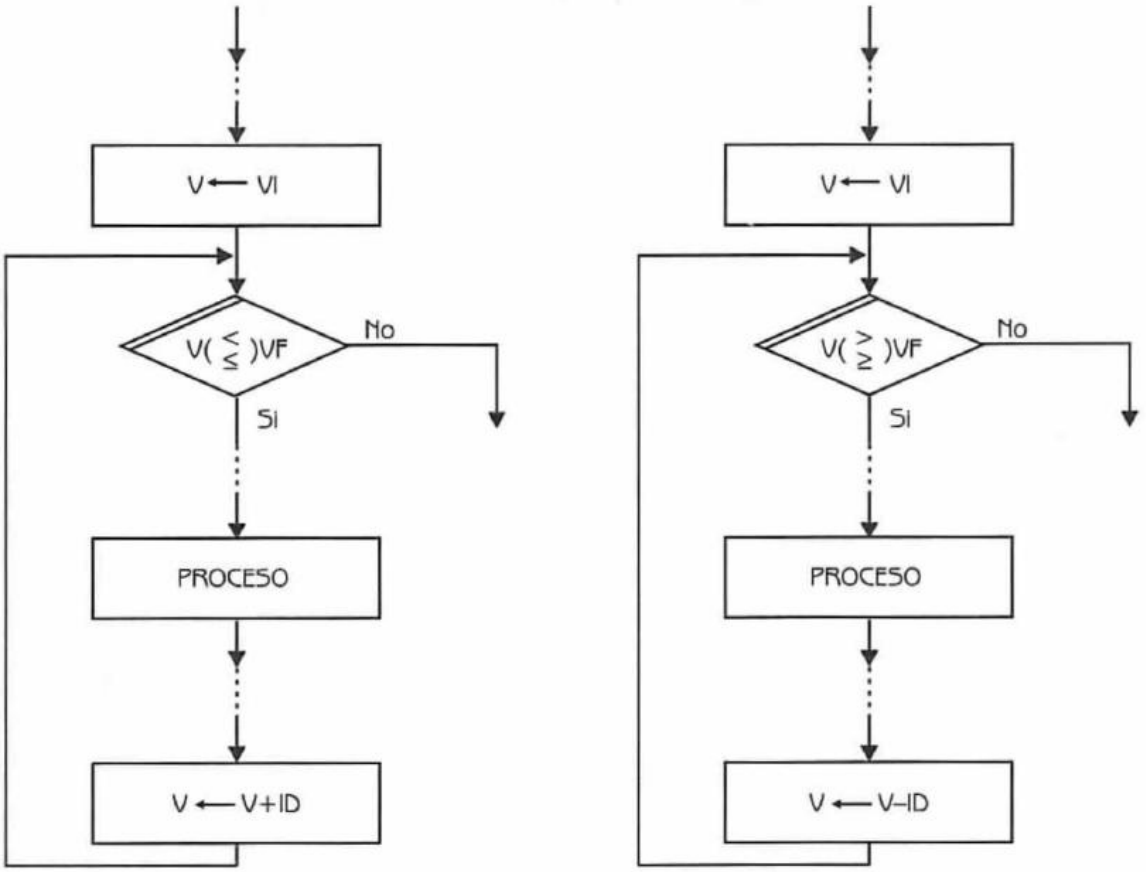
3.2 La estructura repetitiva *repetir* (FOR)

La estructura *repetir* conocida comúnmente como FOR, es la estructura algorítmica adecuada para utilizar en un ciclo que se ejecutará un número definido de veces. Este tipo de estructura está presente en todos los lenguajes de programa-

www.FreeLibros.com

ción, ya sean estructurados u orientados a objetos. Por ejemplo cuando necesitamos calcular la nómina total de la empresa, tenemos que sumar los sueldos de los N empleados de la misma. Cuando necesitamos obtener el promedio de calificaciones de un curso, debemos sumar las N calificaciones de los alumnos y dividir esa suma entre N. Es decir, sabemos de antemano cuántas veces tenemos que repetir una determinada operación, acción o tarea. El número de repeticiones *no* depende de las proposiciones dentro del ciclo. El número de veces se obtiene del planteamiento del problema o de una lectura que indica que el número de iteraciones se debe realizar para N ocurrencias.

El diagrama de flujo de la estructura algorítmica *repetir* es el siguiente:



a) En forma ascendente

a) En forma descendente

Diagrama de Flujo 3.1

Donde:

V es la variable de control del ciclo.
VI es el valor inicial.
VF es el valor final.
ID es el incremento o decremento, según sea la estructura *repetir* ascendente o descendente.

www.FreeLibros.com

V (contador del ciclo, generalmente representado por las letras I, J, K, V) toma un valor inicial y se compara con VF (valor final). El ciclo se ejecuta mientras V es menor, menor igual, mayor o mayor igual al valor de V. El valor de V se incrementa o decrementa en cada iteración. Cuando V supera el valor de V. entonces el ciclo se detiene.

Observe el lector que la modificación introducida al simbolo de la decisión –doble linea en la parte superior izquierda– en la estructura *repetir*, le será de gran utilidad tanto en la construcción como en la interpretación del diagrama de flujo. Indudablemente, este cambio sencillo en la estructura *repetir* le permitirá diferenciarlo de la otra estructura repetitiva: *mientras*.

En lenguaje algorítmico, al Diagrama de Flujo 3.1(a), lo expresamos de esta forma:

Programa 3.1

```
.
.
.
Hacer V ←—— VI
Repetir con V desde VI hasta VF.
    .
    .
    .
      {proceso}
    .
    .
    .
      Hacer V ←—— V + ID
{Fin del ciclo}
```

Veamos a continuación el siguiente ejemplo:

Ejemplo 3.1

Construya un diagrama de flujo tal que dado como datos los sueldos de los 10 trabajadores de una empresa, obtenga el total de nómina de la misma. Considere además que no puede utilizar estructuras algorítmicas repetitivas en la solución del problema.

Datos: SUE1, SUE2, SUE3, SUE4, SUE5, SUE6, SUE7, SUE8, SUE9, SUE10.

www.FreeLibros.com

Donde:

SUE1, SUE2,..., SUE10 son variables de tipo real que representan los sueldos de los 10 trabajadores.

A continuación presentamos la solución correspondiente del problema:

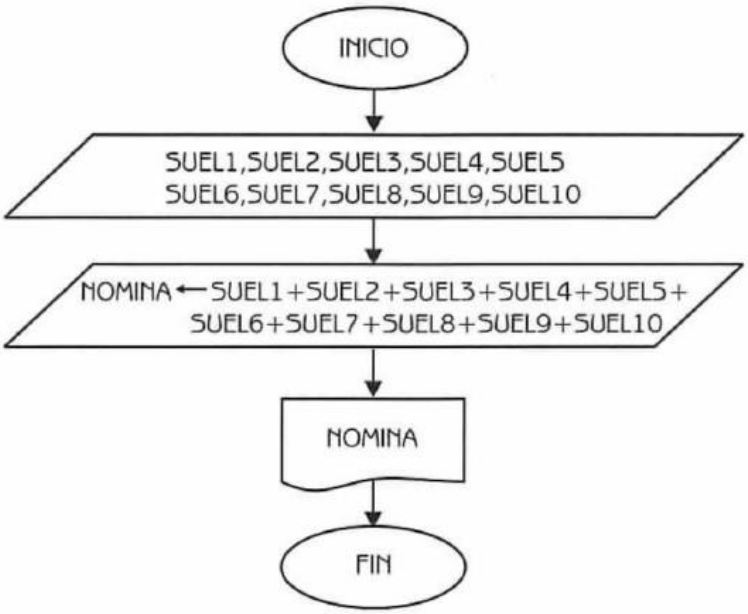


Diagrama de Flujo 3.2

Explicación de las variables

SUE1, SUE2, SUE3, SUE4, SUE5, SUE6, SUE7, SUE8, SUE9 y *SUE10*: Variables de tipo real.

NOMINA: Variable de tipo real. Almacena la suma de los sueldos de todos los trabajadores.

Considere el lector qué sucedería si en lugar de tener 10 empleados, la empresa tuviera 100 o 1000. En realidad el problema es sencillo y se puede resolver con una estructura algorítmica repetitiva, de tal forma que el ciclo se ejecute tantas veces como empleados tenga la empresa. Observemos el siguiente ejemplo:

Ejemplo 3.2

A continuación presentamos la solución del problema anterior, utilizando una estructura algorítmica repetitiva

Datos: $SUE_1, SUE_2,...,SUE_{10}$

www.FreeLibros.com

Donde:

SUE_i es una variable de tipo real que representa el sueldo del trabajador i ($1 \leq i \leq 10$).

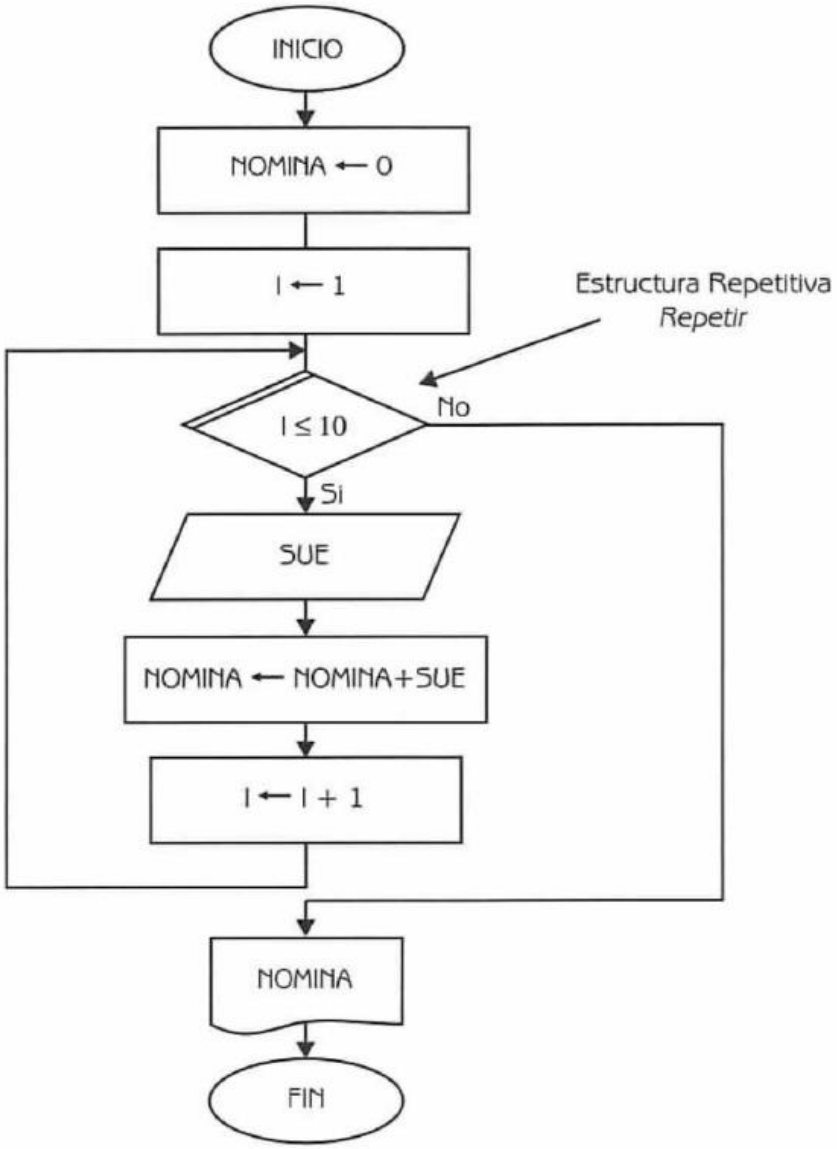


Diagrama de Flujo 3.3

Explicación de las variables

I: Es una variable de tipo entero que representa la variable de control del ciclo. Contabiliza el número de veces que ha de repetirse una determinada acción. El contador toma un valor inicial (generalmente 0 o 1) y se incrementa en la mayoría de los casos en una unidad en cada vuelta del ciclo.

NOMINA: Es una variable de tipo real que representa un *acumulador*. Este se utiliza cuando debemos obtener el total acumulado de un conjunto de cantidades. Generalmente se inicializa en cero.

www.FreeLibros.com

SUE: Es una variable de tipo real. Representa el sueldo del trabajador.

En la siguiente tabla podemos observar el seguimiento del algoritmo para los siguientes datos (SUE): $1 500, $890, $700, $950, $2 300, $1 650, $1 800, $1 400, $760, $900.

Inicia el Ciclo ⟶

Finaliza el Ciclo ⟶

Tabla 3.1

I	SUE	NOMINA
1		0
2	1500	1500
3	890	2390
4	700	3090
5	950	4040
6	2300	6340
7	1650	7990
8	1800	9790
9	1400	11190
10	760	11950
11	900	12850

: Expresa la cantidad que se imprime.

Nota: Observe el lector que cuando inicia el ciclo se lee el sueldo (SUE), se actualiza la nómina (NOMINA) y se incrementa la variable I (contador del ciclo). Este es el criterio que utilizaremos en todo el capítulo para la construcción de tablas

A continuación presentamos la solución del problema, en lenguaje algorítmico.

Programa 3.2

```
NOMINA
{El programa calcula el total de la nómina de un grupo de 10 empleados}
{I es una variable de tipo entero. SUE y NOMINA son variables de tipo real}
1. Hacer NOMINA ⟵ 0 e I ⟵ 1
2. Repetir con I desde 1 hasta 10
                Leer SUE
                Hacer   NOMINA ⟵ NOMINA + SUE e I ⟵ I + 1
3. {Fin del ciclo del paso 2}
4. Escribir NOMINA
```

www.FreeLibros.com

Observemos a continuación otro ejemplo.

Ejemplo 3.3

Escriba un diagrama de flujo tal que dado como datos N números enteros, obtenga el número de ceros que hay entre estos números.

Datos: N, NUM_1, NUM_2,..., NUM_N

Donde:

N es una variable de tipo entero que representa el número de datos que se ingresan.

NUM_i es una variable de tipo entero que representa al número i ($1 \leq i \leq N$).

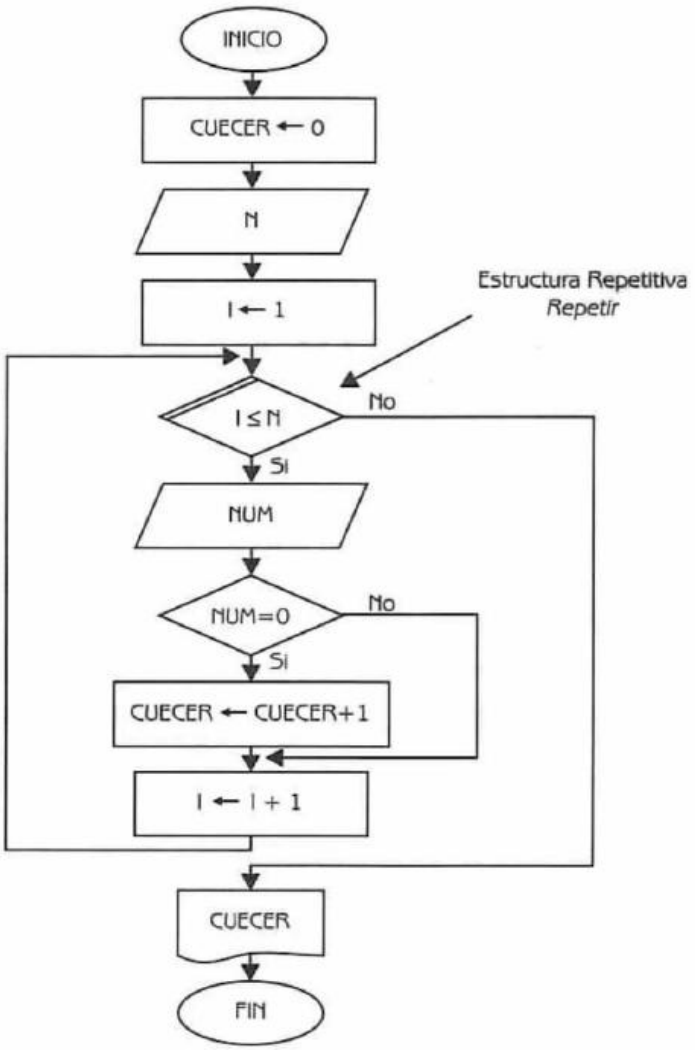


Diagrama de Flujo 3.4

www.FreeLibros.com

Explicación de las variables

I: Variable de tipo entero. Representa al contador del ciclo.

N y NUM: Variables de tipo entero.

CUECER: Variable de tipo entero. Cuenta el número de ceros.

A continuación en la siguiente tabla podemos observar el seguimiento del algoritmo.

Inicia el Ciclo ⟶

Finaliza el Ciclo ⟶

Tabla 3.2

I	N	CUECER	NUM
1	12	0	
2			18
3			23
4		1	0
5			17
6			22
7		2	0
8			37
9			43
10		3	0
11			27
12			41
13			53

: Expresa el valor que se imprime.

Programa 3.3

```
CUENTA _CEROS

{El programa cuenta cuántos ceros hay en un grupo de N números enteros}

{I, N, NUM, y CUECER son variables de tipo entero}

1. Hacer CUECER ⟵ 0
2. Leer N
3. Hacer I ⟵ 1
4. Repetir con I desde 1 hasta N
        Leer NUM
```

www.FreeLibros.com

```
    4.1 Si NUM = 0 entonces
            Hacer CUECER ←— CUECER + 1
    4.2 {Fin del condicional del paso 4.1}
        Hacer I ←— I + 1
5. Fin del ciclo del paso 4
6. Escribir CUECER
```

3.3 La estructura repetitiva *mientras* (WHILE)

La estructura algorítmica *mientras*, comúnmente conocida como while, es la estructura adecuada para utilizar en un ciclo cuando *no* sabemos el número de veces que éste se ha de repetir. Dicho número depende de las proposiciones dentro del ciclo. Ejemplos en la vida cotidiana encontramos muchos. Por ejemplo, supongamos que tenemos que obtener el total de una serie de gastos, pero no sabemos exactamente cuántos son; o cuando tenemos que sacar el promedio de calificaciones de un examen, pero no sabemos precisamente cuántos alumnos lo aplicaron. Tenemos que sumar las calificaciones e ir contando el número de alumnos, esto con el fin de poder obtener posteriormente el promedio. El ciclo se repite mientras tengamos calificaciones de alumnos.

En la estructura *mientras* se distinguen dos partes:

- *Ciclo*: Conjunto de instrucciones que se ejecutarán repetidamente.
- *Condición de terminación*: La evaluación de esta condición permite decidir cuándo finalizará la ejecución del ciclo. La condición se evalúa al inicio del mismo.

El diagrama de flujo de la estructura algorítmica *mientras* es el siguiente:

Donde

PI: La proposición inicial, debe tener un valor verdadero inicialmente. Si el valor de PI es falso, entonces el ciclo no se ejecuta.

www.FreeLibros.com

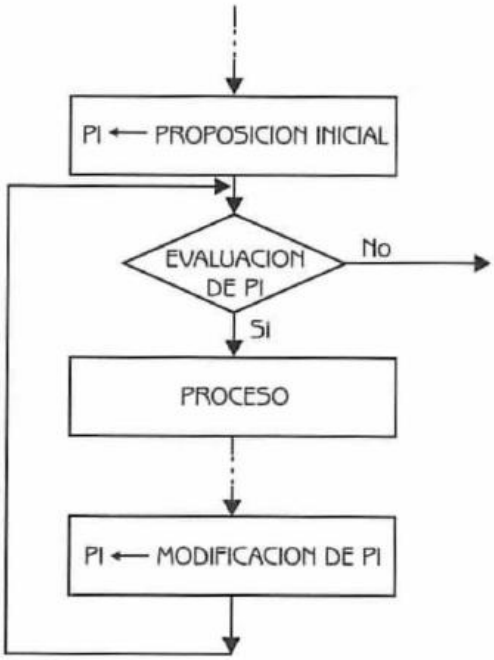


Diagrama de Flujo 3.5

Debe existir también un enunciado dentro del ciclo que afecte la condición, para evitar que el ciclo se ejecute indefinidamente.

En lenguaje algorítmico la estructura *mientras* la expresamos de esta forma:

Programa 3.4

```
.
.
.
Hacer      PI ←—— Proposición inicial
Mientras      PI es verdadero repetir
      .
      .
      .
      {PROCESO}
      .
      .
      .
      Hacer      PI ←—— modificación de PI
{Fin del ciclo}
```

Veamos a continuación el siguiente ejemplo:

www.FreeLibros.com

Ejemplo 3.4

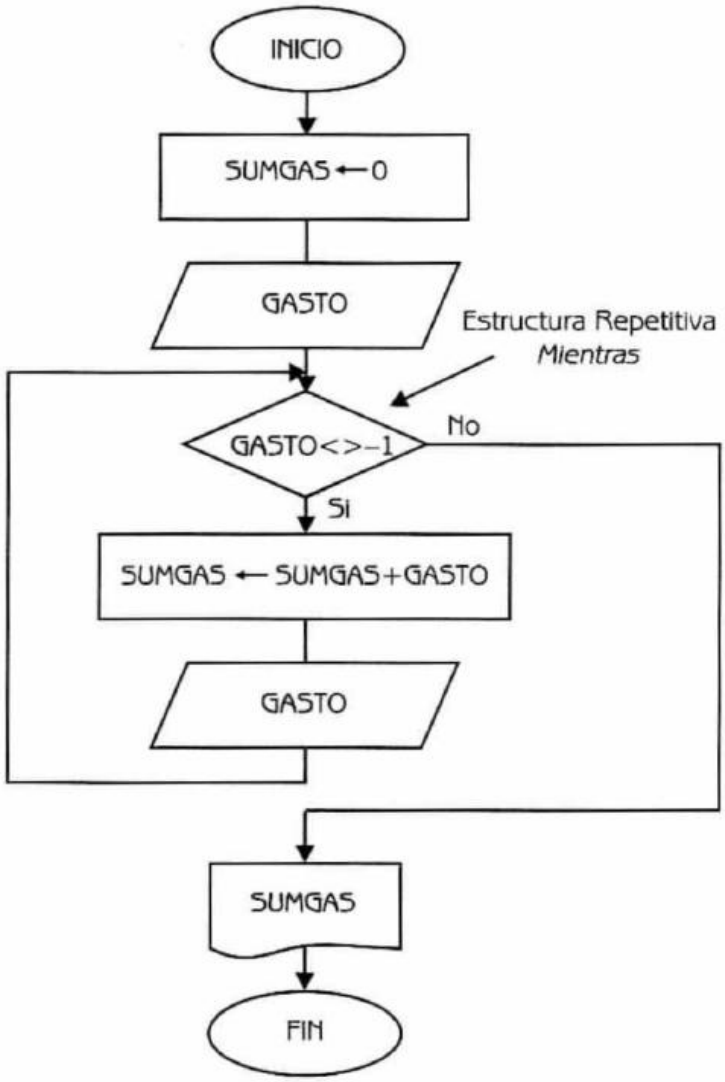
Supongamos que debemos obtener la suma de los gastos que hicimos en nuestro último viaje, pero no sabemos exactamente cuántos fueron. Los datos son expresados en forma:

Datos: $GASTO_1, GASTO_2, \ldots, -1$

Donde:

$GASTO_i$ es una variable de tipo real que representa el gasto número i.

A continuación presentamos la solución del problema.



Explicación de las variables

SUMGAS: Es una variable de tipo real.
Es un acumulador.
Acumula los datos efectuados.

GASTO: Es una variable de tipo real. Su valor en la primera lectura debe ser verdadero, es decir distinto de –1. Su valor se modifica en cada vuelta del ciclo. Cuando gasto tome el valor de –1, entonces el ciclo se detendrá.

Diagrama de Flujo 3.6

www.FreeLibros.com

Nota: Observe el lector que en este tipo de problemas deberemos realizar dos lecturas, una antes de iniciar el ciclo y otra antes de finalizar el mismo.

En la siguiente tabla podemos observar el seguimiento del algoritmo para los siguientes datos (GASTO): $2,528, $3,500, $1,600, $1,850, $150, –1

Tabla 3.3

	SUMGAS	GASTO
Inicia el Ciclo ⟶	0	2528
	2528	3500
	6028	1600
	7628	1850
	9478	150
Finaliza el Ciclo ⟶	9628	–1

: Expresa el valor que se imprime

Programa 3.5

```
SUMA_GASTOS

{El programa obtiene el total de gastos de un viaje}

{Gasto y SUMGAS son variables de tipo real}

1. Hacer SUMGAS ⟵ 0
2. Leer GASTO
3. Mientras GASTO < > –1 Repetir
        Hacer SUMGAS ⟵ SUMGAS + GASTO
        Leer GASTO
4. {Fin del ciclo del paso 3}
5. Escribir SUMGAS
```

Observemos a continuación otro ejemplo.

Ejemplo 3.5

Escriba un diagrama de flujo tal que dado un grupo de número naturales positivos, calcule e imprima el cubo de estos números.

www.FreeLibros.com

Datos: NUM_1, NUM_2, NUM_3,..., –1 (NUMi es una variables de tipo entero que representa el número positivo i. El fin de datos está dado por –1).

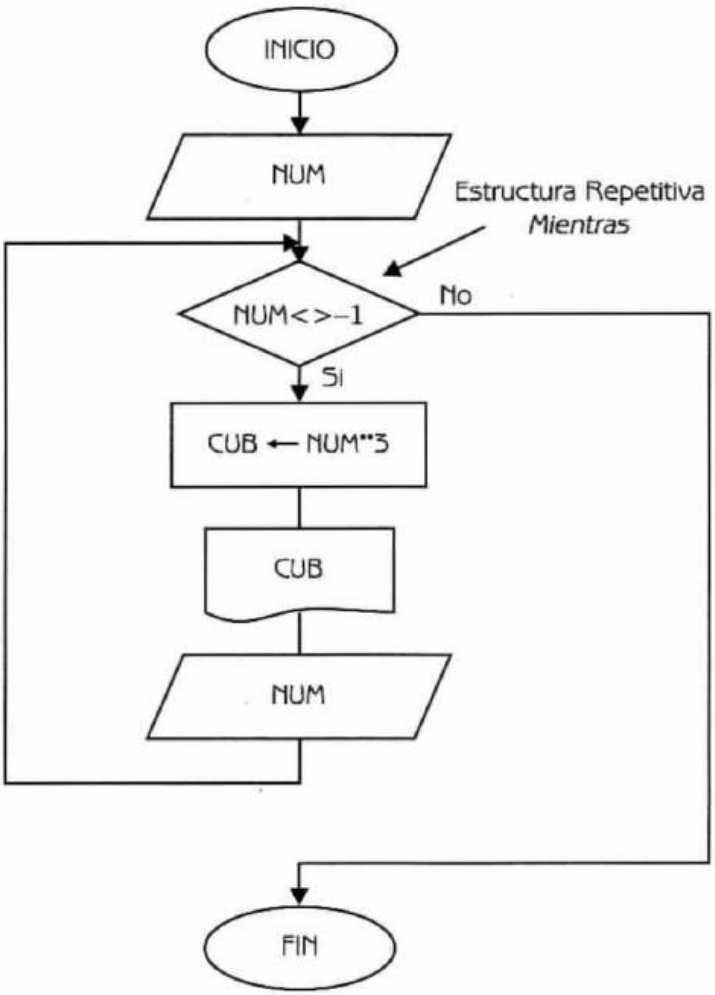


Diagrama de Flujo 3.7

Explicación de las variables

NUM : Variable de tipo entero.

CUB : Variable de tipo real. Almacena el cubo del número que se ingresa.

En la siguiente tabla podemos observar el seguimiento del diagrama de flujo.

www.FreeLibros.com

Inicio el Ciclo ⟶

Finaliza el Ciclo ⟶

Tabla 3.4	
NUM	CUB
5	
13	125
7	2197
48	343
18	110592
27	5832
94	19683
62	830584
114	238328
−1	1481544

: Expresa valores que se imprimen.

Programa 3.6

```
CUBO

{El programa calcula el cubo de un grupo de números enteros positivos que se
ingresan}

{NUM es una variable de tipo entero. CUB es una variable de tipo real}

1. Leer NUM
2. Mientras (NUM < > −1) Repetir
            Hacer CUB ⟵ NUM ** 3
            Escribir CUB
            Leer NUM
3. {Fin del ciclo del paso 2}
```

A continuación presentamos una serie de problemas diseñados expresamente como elementos de ayuda para el análisis y la retención de los conceptos. Además se utilizan en muchos de ellos, tablas que permiten hacer un seguimiento del algoritmo.

www.FreeLibros.com

Problemas resueltos

Problema 3.1

Escriba un diagrama de flujo tal que dado como datos 270 números enteros, obtenga la suma de los números impares y el promedio de los números pares.

Datos: $NUM_1, NUM_2, \ldots, NUM_{270}$

Donde:

NUM_i es una variable de tipo entero que representa el número entero i que se ingresa ($1 \leq i \leq 270$).

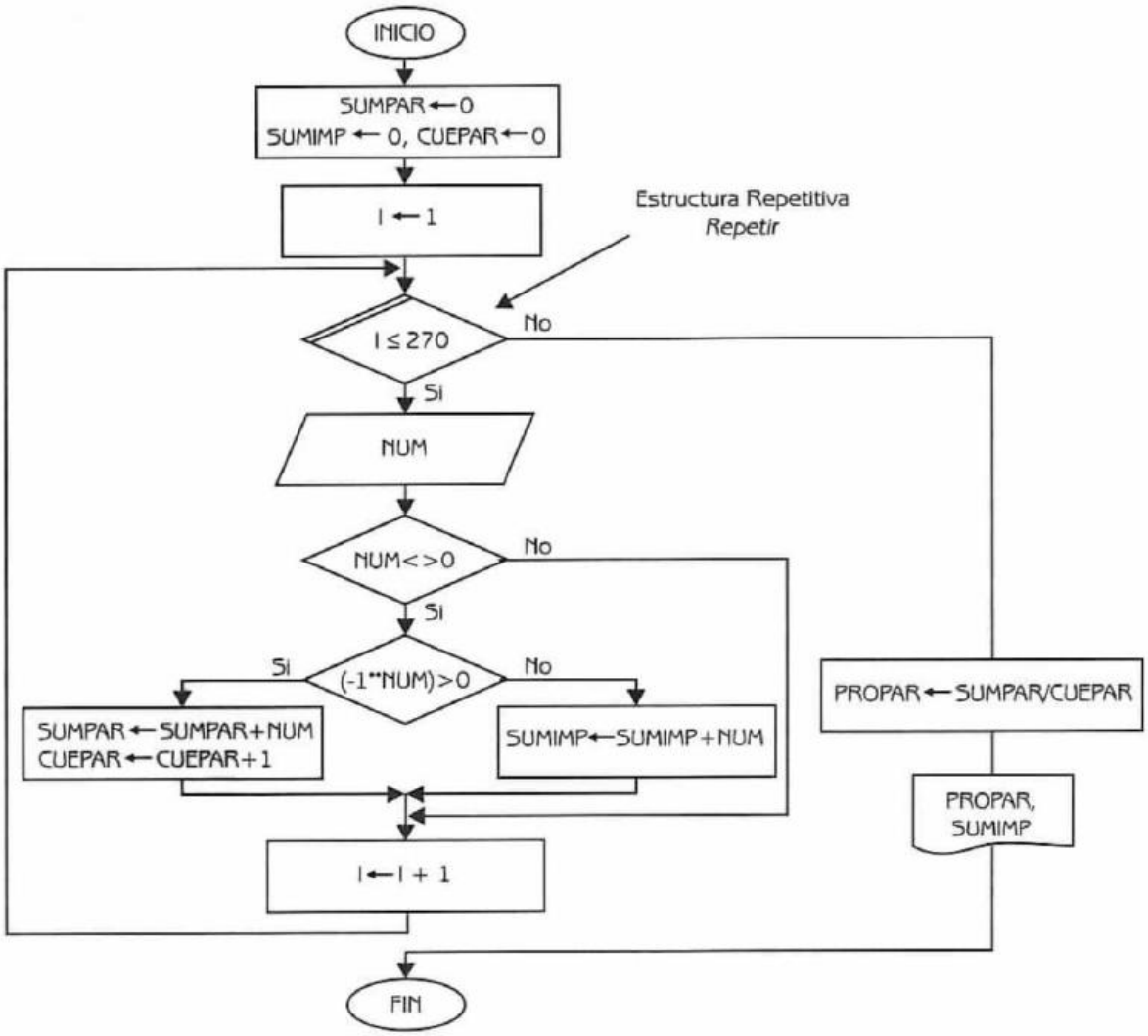


Diagrama de Flujo 3.8

www.FreeLibros.com

Explicación de las variables

I:	Variable de tipo entero. Representa la variable de control del ciclo.
NUM:	Variable de tipo entero.
SUMPAR y *SUMIMP*:	Variables de tipo real. Acumulan los números pares e impares, respectivamente.
CUEPAR:	Variable de tipo entero. Cuenta el número de números pares.
PROPAR:	Variable de tipo real. Almacena el promedio de los números pares.

A continuación presentamos el diagrama de flujo en lenguaje algorítmico.

Programa 3.7

```
PARES_E_IMPARES

{El programa dados 270 números enteros, obtiene la suma de los números
impares y el promedio de los números pares}

{I, NUM y CUEPAR son variables de tipo entero. SUMPAR, SUMIMP y PROPAR son
variables de tipo real}

1. Hacer SUMPAR ←— 0, SUMIMP ←— 0, CUEPAR ←— 0 e I ←— 1
2. Repetir con I desde 1 hasta 270
        Leer NUM
   2.1 Si (NUM < > 0) entonces
       2.1.1 Si (–1 ** NUM ) > 0
             entonces
               Hacer SUMPAR ←— SUMPAR + NUM y CUEPAR ←— CUEPAR + 1
             sino
               Hacer SUMIMP ←— SUMIMP + NUM
       2.1.2 {Fin de condicional del paso 2.1.1}
   2.2 {Fin del condicional del paso 2.1}
       Hacer I ←— I + 1
3. {Fin del ciclo del paso 2}
4. Hacer PROPAR ←— SUMPAR / CUEPAR
5. Escribir PROPAR y SUMIMP
```

Problema 3.2

Escriba un diagrama de flujo que obtenga la suma e imprima los términos de la siguiente serie:

2, 5, 7, 10, 12, 15, 17, . . ., 1 800

www.FreeLibros.com

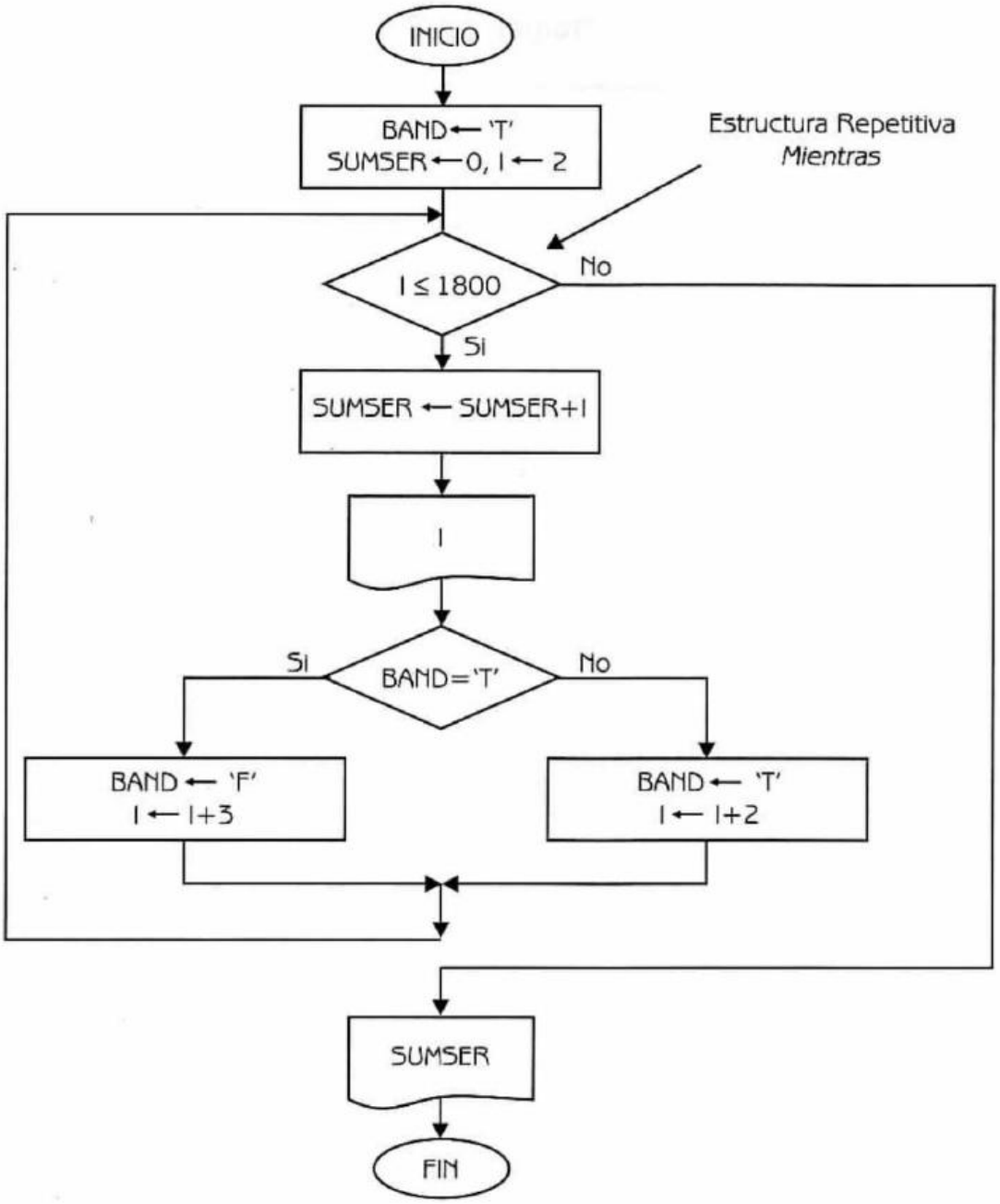


Diagrama de Flujo 3.9

Explicación de las variables

I: Variable de tipo entero. Se utiliza para incrementar el valor de los términos de la serie.

SUMSER: Variable de tipo entero. Acumula los términos de la serie.

BAND: Variable de tipo caracter. Es una variable auxiliar que se utiliza para indicar si al siguiente término de la serie hay que sumarle 3 o 2.

A continuación presentamos el diagrama de flujo en lenguaje algorítmico

www.FreeLibros.com

Programa 3.8

```
SUMA_E_IMPRIME_SERIE

{El programa imprime y suma los términos de una serie}

{I y SUMSER son variables de tipo entero. BAND es una variable de tipo caracter}

1. Hacer SUMSER ←— 0, BAND ←— 'T' e I ←— 2
2. Mientras (I ≤ 1800) Repetir
        Hacer SUMSER ←— SUMSER + I
        Escribir I
   2.1 Si BAND = 'T'
            entonces
                Hacer BAND ←— 'F' e I ←— I + 3
            sino
                Hacer BAND ←— 'T' e I ←— I + 2
   2.2 {Fin del condicional del paso 2.1}
3. {Fin del ciclo del paso 2}
4. Escribir SUMSER
```

Problema 3.3

Escriba un diagrama de flujo que lea un número entero N y calcule el resultado de la siguiente serie:

$$1 - \frac{1}{2} + \frac{1}{3} - \frac{1}{4} + \cdots \pm \frac{1}{N}$$

Datos: N (variable de tipo entero que representa el número de términos de la serie).

www.FreeLibros.com

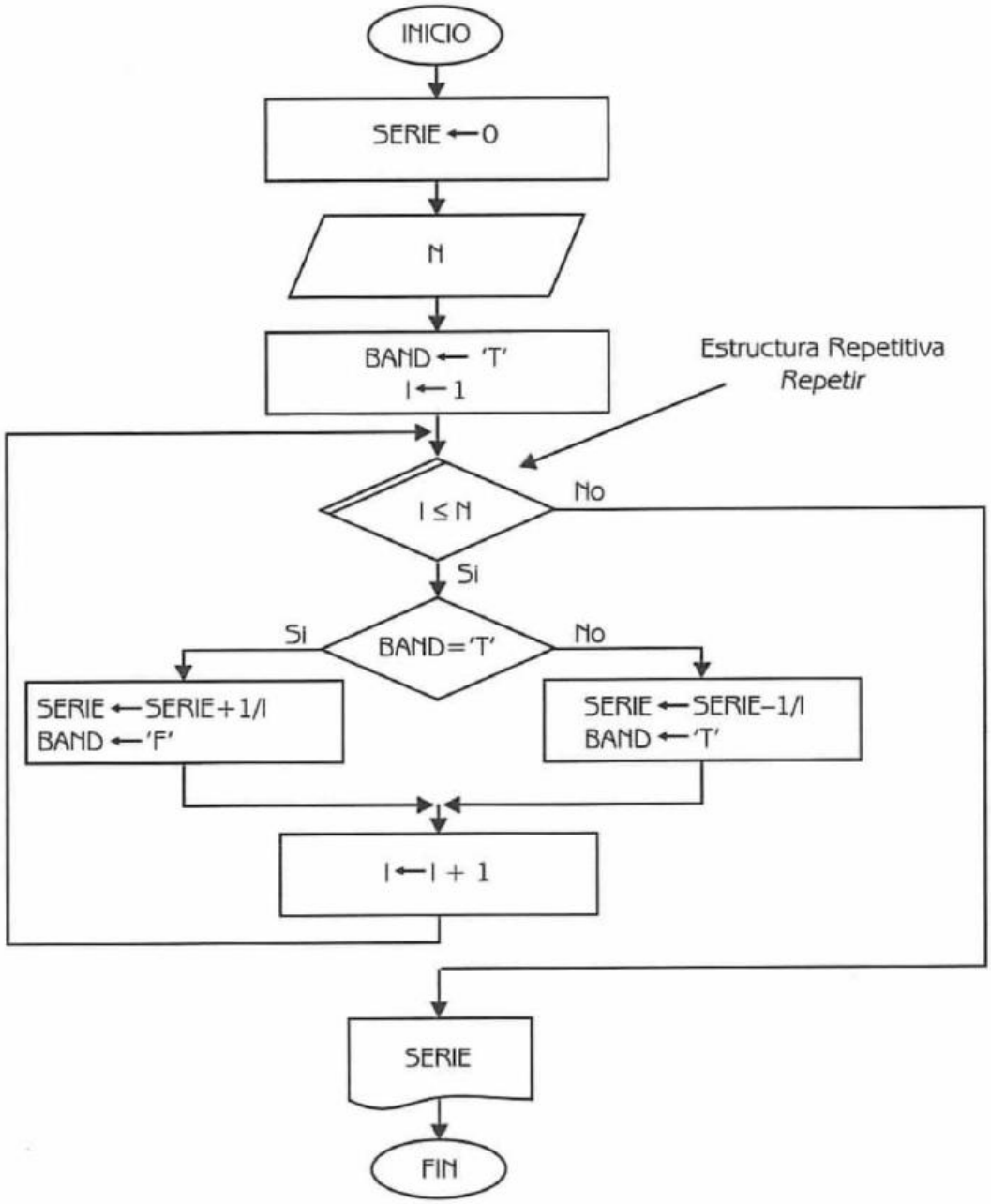


Diagrama de Flujo 3.10

Explicación de las variables

I: Variable de tipo entero. Representa la variable de control del ciclo.
N: Variable de tipo entero.
SERIE: Variable de tipo real. Acumula el resultado de los términos de la serie.
BAND: Variable de tipo caracter. Es una variable auxiliar que nos ayuda a determinar si debemos sumar o restar el término de la serie.

A continuación en la siguiente tabla podemos observar el comportamiento del algoritmo.

www.FreeLibros.com

Tabla 3.5

I	N	BAND	SERIE
1	9	'T'	0
2		'F'	1
3		'T'	0.5
4		'F'	0.8333
5		'T'	0.5833
6		'F'	0.7833
7		'T'	0.6166
8		'F'	0.7594
9		'T'	0.6344
10		'F'	0.7456

Inicia el Ciclo ⟶

Fin del Ciclo ⟶

: Expresa valores que se imprimen.

Programa 3.9

```
SERIE_TRES

{El programa calcula el resultado de una determinada serie}

{I y N son variables de tipo entero. SERIE es una variable de tipo real. BAND es una
variable de tipo carácter}

1. Hacer SERIE ⟵ 0
2. Leer N
3. Hacer BAND ⟵ 'T' e I ⟵ 1
4. Repetir con I desde 1 hasta N
   4.1 Si BAND = 'T'
            entonces
                 Hacer SERIE ⟵ SERIE + 1 / I y BAND ⟵ 'F'
            sino
                 Hacer SERIE ⟵ SERIE – 1 / I y BAND ⟵ 'T'
   4.2 {Fin del condicional del paso 4.1}
       Hacer I ⟵ I + 1
5. {Fin del ciclo del paso}
6. Escribir SERIE
```

Problema 3.4

Calcule el aumento de sueldo para un grupo de empleados de una empresa teniendo en cuenta el siguiente criterio:

www.FreeLibros.com

Si el sueldo es inferior a $1 000 : Aumento 15%
Si el sueldo es mayor o igual a $1 000 : Aumento 12%

Imprima el sueldo nuevo del trabajador y el total de nómina de la empresa, considerando este nuevo aumento.

Datos: $SUE_1, SUE_2, \ldots, -1$

Donde:

SUEi es una variable de tipo real que representa el sueldo del trabajador i. El fin de datos se expresa con –1.

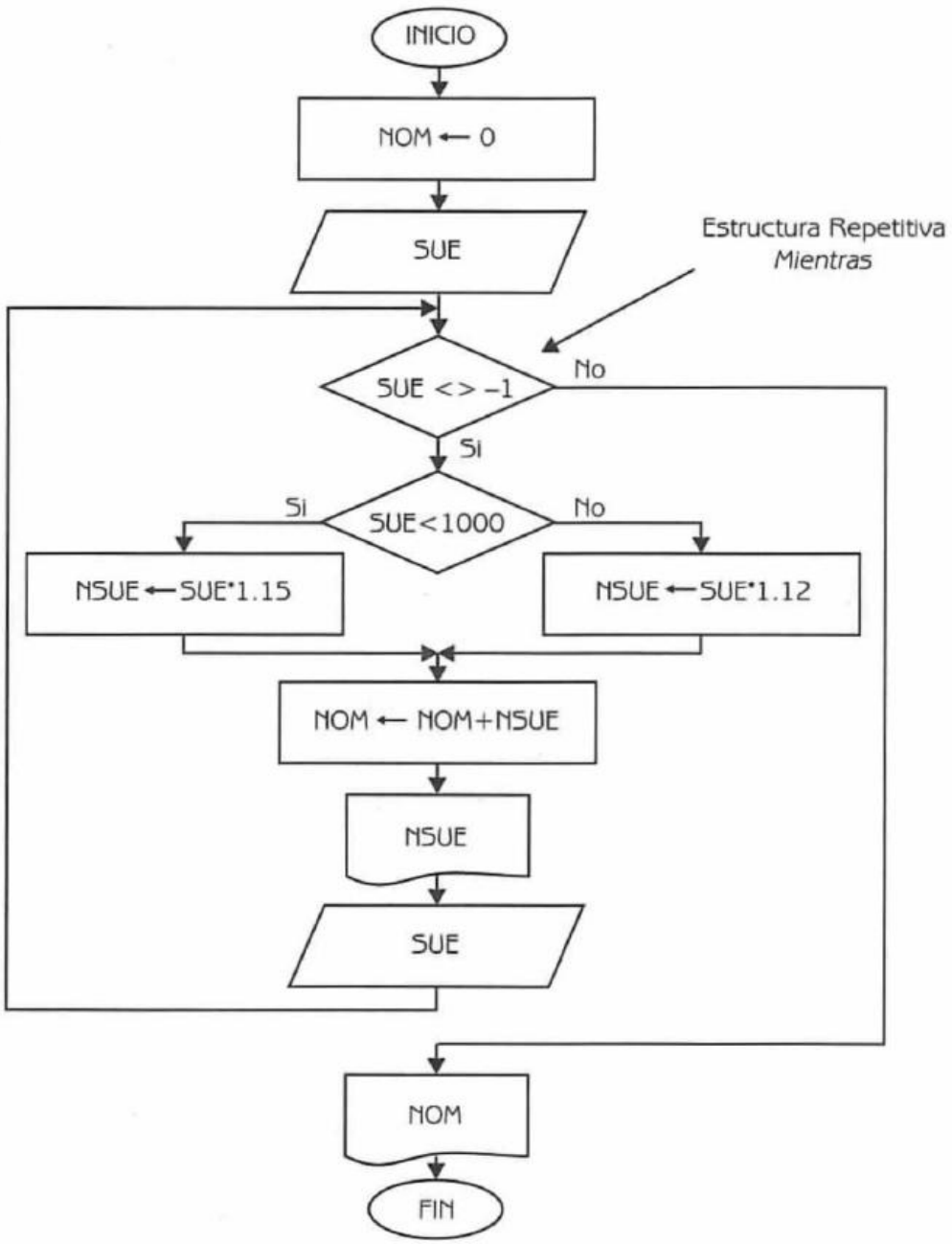


Diagrama de Flujo 3.11

www.FreeLibros.com

Explicación de las variables

NOM: Variable de tipo real. Acumula el nuevo sueldo de los trabajadores.

SUE: Variable de tipo real.

NSUE: Variable de tipo real. Expresa el nuevo sueldo del trabajador.

A continuación en la tabla 3.6, podemos observar el seguimiento del algoritmo.

Tabla 3.6

	NOM	SUE	NSUE
	0	855	
Inicia el Ciclo ⟶	983.25	760.32	983.25
	1857.61	1100.20	874.36
	3089.83	614	1232.22
	3795.93	2600	706.10
	6707.93	817.50	2912.00
	7648.05	1280.30	940.12
	9081.98	687	1433.93
Fin del Ciclo ⟶	9872.03	−1	790.05

: Expresa valores que se imprimen.

Programa 3.10

```
AUMENTO_SUELDO

{El programa, teniendo en cuenta ciertos criterios, calcula el aumento de sueldo
para un grupo de trabajadores. Imprime el nuevo sueldo del trabajador y la nómina
correspondiente}

{NOM, SUE y NSUE son variables de tipo real}

1. Hacer NOM ⟵ 0
2. Leer SUE
3. Mientras (SUE < > −1) Repetir
```

www.FreeLibros.com

```
    3.1 Si SUE < 1000
            entonces
                Hacer NSUE ←—— SUE * 1.15
            sino
                Hacer NSUE ←—— SUE * 1.12
    3.2 {Fin del condicional del paso 3.1}
        Hacer NOM ←—— NOM + NSUE
        Escribir NSUE
        Leer SUE
4. {Fin del ciclo del paso 3}
5. Escribir NOM
```

Problema 3.5

Dado N números enteros como dato haga un diagrama de flujo que:

a) Obtenga cuántos números leídos fueron mayores que cero.

b) Calcule el promedio de los números positivos.

c) Obtenga el promedio de todos los números.

Datos: $N, NUM_1, NUM_2, \ldots, NUM_N$

Donde:

N es una variable de tipo entero que representa el número de datos que vamos a ingresar.

NUM_i es una variable de tipo entero que representa el número i que se ingresa ($1 \leq i \leq N$).

www.FreeLibros.com

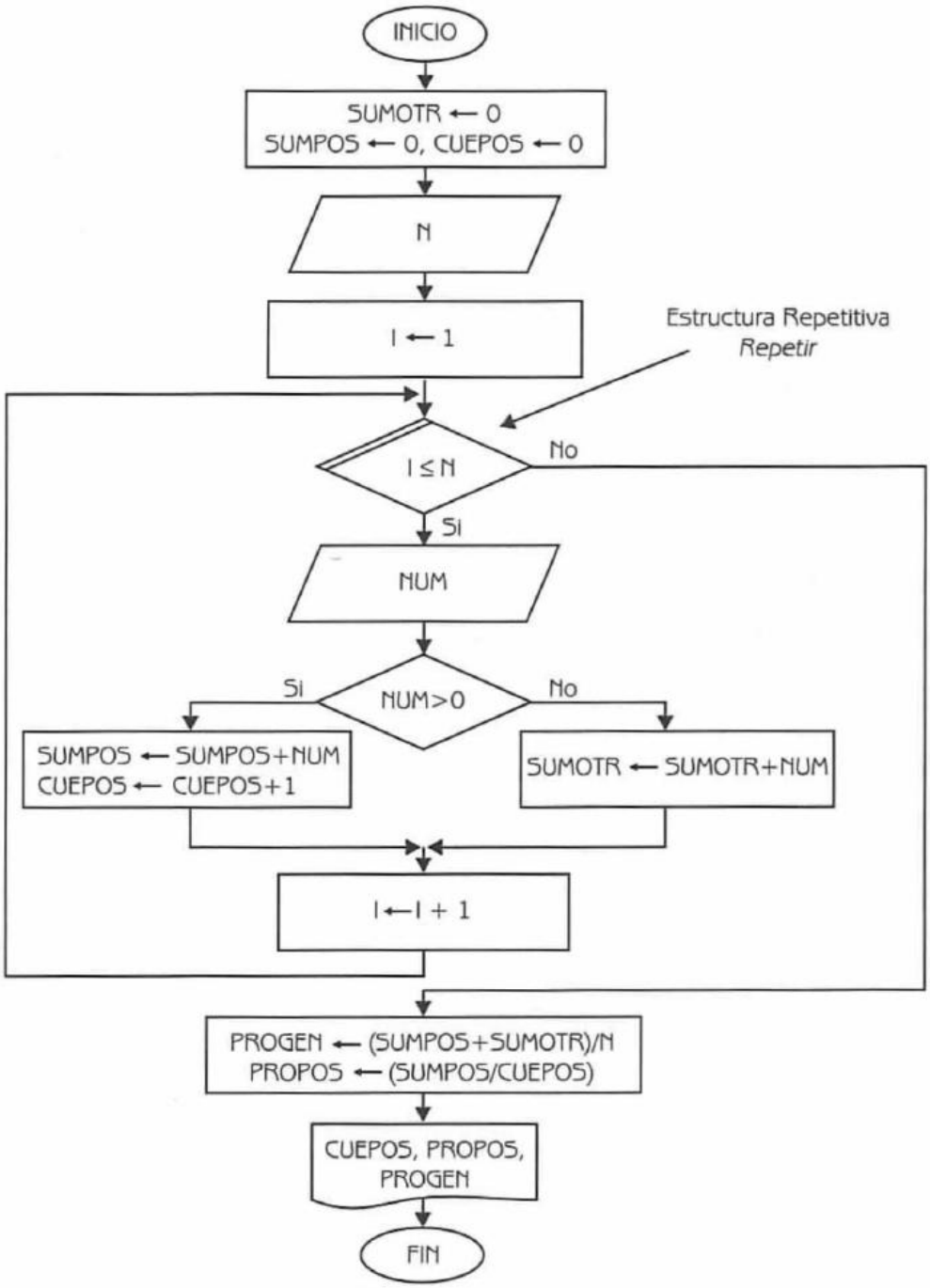


Diagrama de Flujo 3.12

Explicación de las variables

I: Variable de tipo entero. Representa la variable de control del ciclo.
SUMPOS: Variable de tipo real. Acumula los números positivos.
SUMOTR: Variable de tipo real. Acumula los números que no son positivos.
CUEPOS: Variable de tipo entero. Cuenta los números positivos.
N: Variable de tipo entero.

www.FreeLibros.com

NUM: Variable de tipo entero.

PROGEN: Variable de tipo real. Almacena el promedio general de los números.

PROPOS: Variable de tipo real. Almacena el promedio de los números positivos.

A continuación en la siguiente tabla podemos observar el seguimiento del algoritmo.

Tabla 3.7

	I	N	NUM	SUMPOS	CUEPOS	SUMOTR	PROGEN	PROPOS
Inicio del Ciclo →	1	20		0	0	0		
	2		7	7	1			
	3		12	19	2			
	4		0			0		
	5		6	25	3			
	6		11	36	4			
	7		-3			-3		
	8		17	53	5			
	9		12	65	6			
	10		19	84	7			
	11		0			-3		
	12		11	95	8			
	13		4	99	9			
	14		-5			-8		
	15		-1			-9		
	16		0			-9		
	17		3	102	10			
	18		24	126	11			
	19		0			-9		
	20		3	129	12			
Fin del Ciclo →	21		-16			-25		
							5.2	10.75

: Expresa valores que se imprimen.

www.FreeLibros.com

Programa 3.11

```
PROMEDIO_NUMEROS

{El programa dado N números enteros como dato, obtiene el número de números
positivos, el promedio de los números positivos y el promedio de todos los núme-
ros}

{I, CUEPOS, NUM y N son variables de tipo entero. SUMPOS, SUMOTR, PROGEN y
PROPOS son variables de tipo real.}

1. Hacer SUMPOS ←— 0, CUEPOS ←— 0 y SUMOTR ←— 0
2. Leer N
3. Hacer I ←— 1
4. Repetir con I desde 1 hasta N
        Leer NUM
   4.1 Si NUM > 0
          entonces
             Hacer SUMPOS ←— SUMPOS + NUM y CUEPOS ←— CUEPOS + 1
          sino
             Hacer SUMOTR ←— SUMOTR + NUM
   4.2 {Fin del condicional del paso 4.1}
       Hacer I ←— I + 1
5. {Fin del ciclo del paso 4}
6. Hacer PROGEN ←— (SUMPOS + SUMOTR)/N y PROPOS ←— SUMPOS/CUEPOS
7. Escribir CUEPOS, PROPOS y PROGEN
```

Problema 3.6

Construya un diagrama de flujo tal que dado N números enteros como dato, calcule el mayor y el menor de estos números.

Datos: N, NUM_1, NUM_2, . . ., NUM_N

Donde:

N es una variable de tipo entero que indica el número de enteros que se ingresan.

NUM_i es una variable de tipo entero que representa el número i que se ingresa ($1 \leq i \leq N$).

www.FreeLibros.com

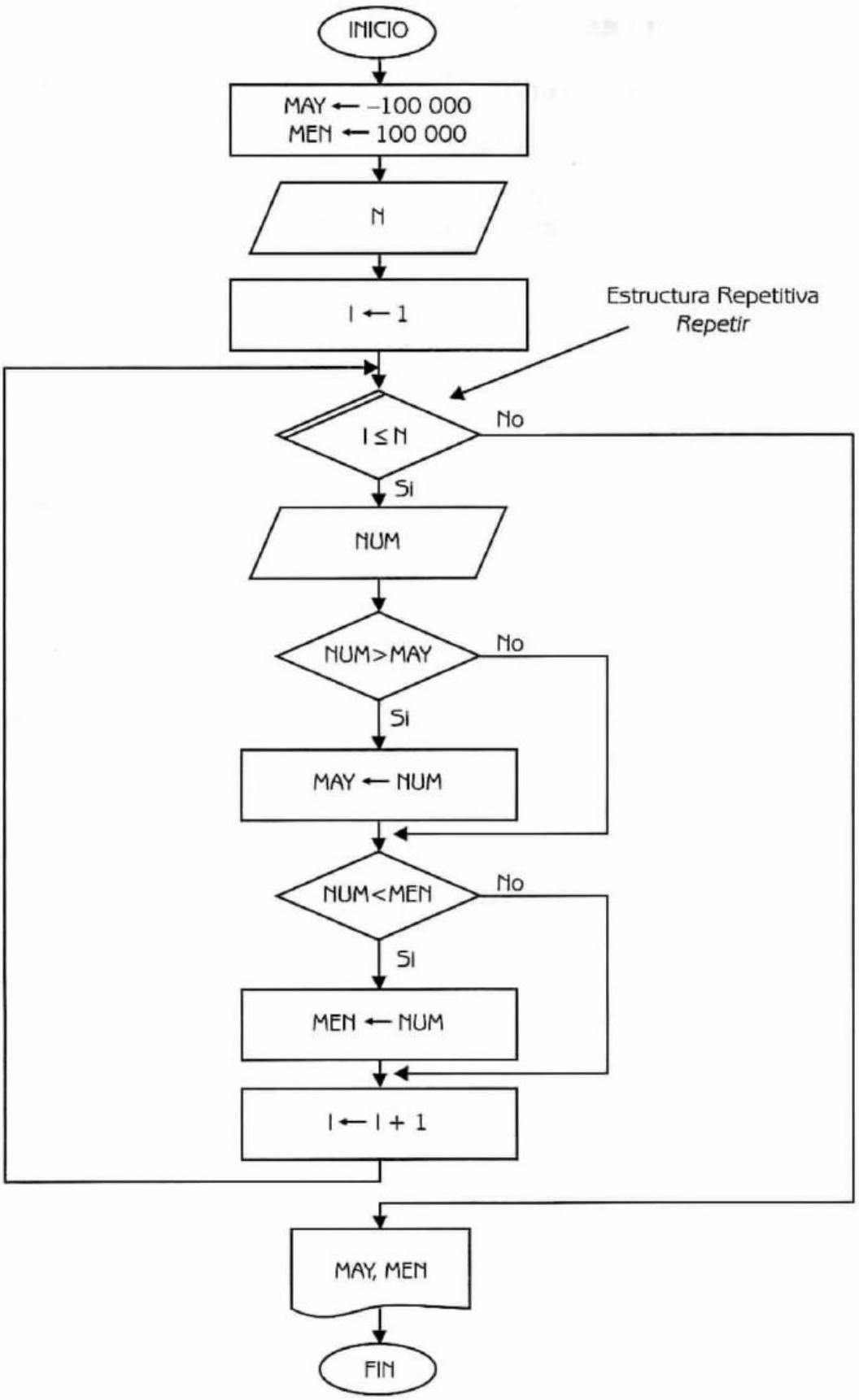


Diagrama de Flujo 3.13

www.FreeLibros.com

Explicación de las variables

I: Variable de tipo entero. Representa la variable de control del ciclo.

N y *NUM*: Variables de tipo entero.

MAY: Variable de tipo entero. Almacena el máximo valor. Se inicializa con un valor negativo alto para que tome el valor del primer número que se ingresa.

MEN: Variable de tipo entero. Almacena el mínimo valor. Se inicializa con un valor alto para que tome el valor del primer número que se ingresa.

Tabla 3.8

	I	N	NUM	MAY	MEN
	1	12		-100 000	100 000
Inicio del Ciclo →	2		170	170	170
	3		44		44
	4		815	815	
	5		1700	1700	
	6		38		38
	7		140		
	8		380		
	9		1020		
	10		116		
	11		14		14
	12		730		
Fin del Ciclo →	13		960		

: Expresa valores que se imprimen.

A continuación presentamos el diagrama de flujo en lenguaje algorítmico.

www.FreeLibros.com

Programa 3.12

```
MAYOR_Y_MENOR

{El programa obtiene el mayor y menor de N números enteros que se ingresan}

{I, N y NUM son variables de tipo entero. MAY y MEN son variables de tipo real}

1. Hacer MAY ←— −100 000 y MEN ←— 100 000
2. Leer N
3. Hacer I ←— 1
4. Repetir con I desde 1 hasta N
        Leer NUM
    4.1 Si NUM > MAY entonces
            Hacer MAY ←— NUM
    4.2 {Fin del condicional del paso 4.1}
    4.3 Si NUM < MEN entonces
            Hacer MEN ←— NUM
    4.4 {Fin del condicional del paso 4.3}
        Hacer I ←— I + 1
5. {Fin del ciclo del paso 4}
6. Escribir MAY y MEN
```

Problema 3.7

Un vendedor ha hecho una serie de ventas y desea conocer aquellas de $200 o menos, las mayores a $200 pero inferiores a $400, y el número de ventas de $400 o superiores a tal cantidad. Haga un diagrama de flujo que le proporcione al vendedor esta información después de haber leído los datos de entrada.

Datos: $N, V_1, V_2, \ldots, V_N$

Donde:

- N es una variable de tipo entero que representa el número de ventas del vendedor.
- V_i es una variable de tipo real que indica la venta i del vendedor $(1 \leq i \leq N)$.

Explicación de las variables

I:	Variable de tipo entero. Representa la variabe de control del ciclo.
CHI, MED y GRA:	Variables de tipo entero. Acumulan el número de ventas menores a $200, a $400 y mayores a $400, respectivamente.

www.FreeLibros.com

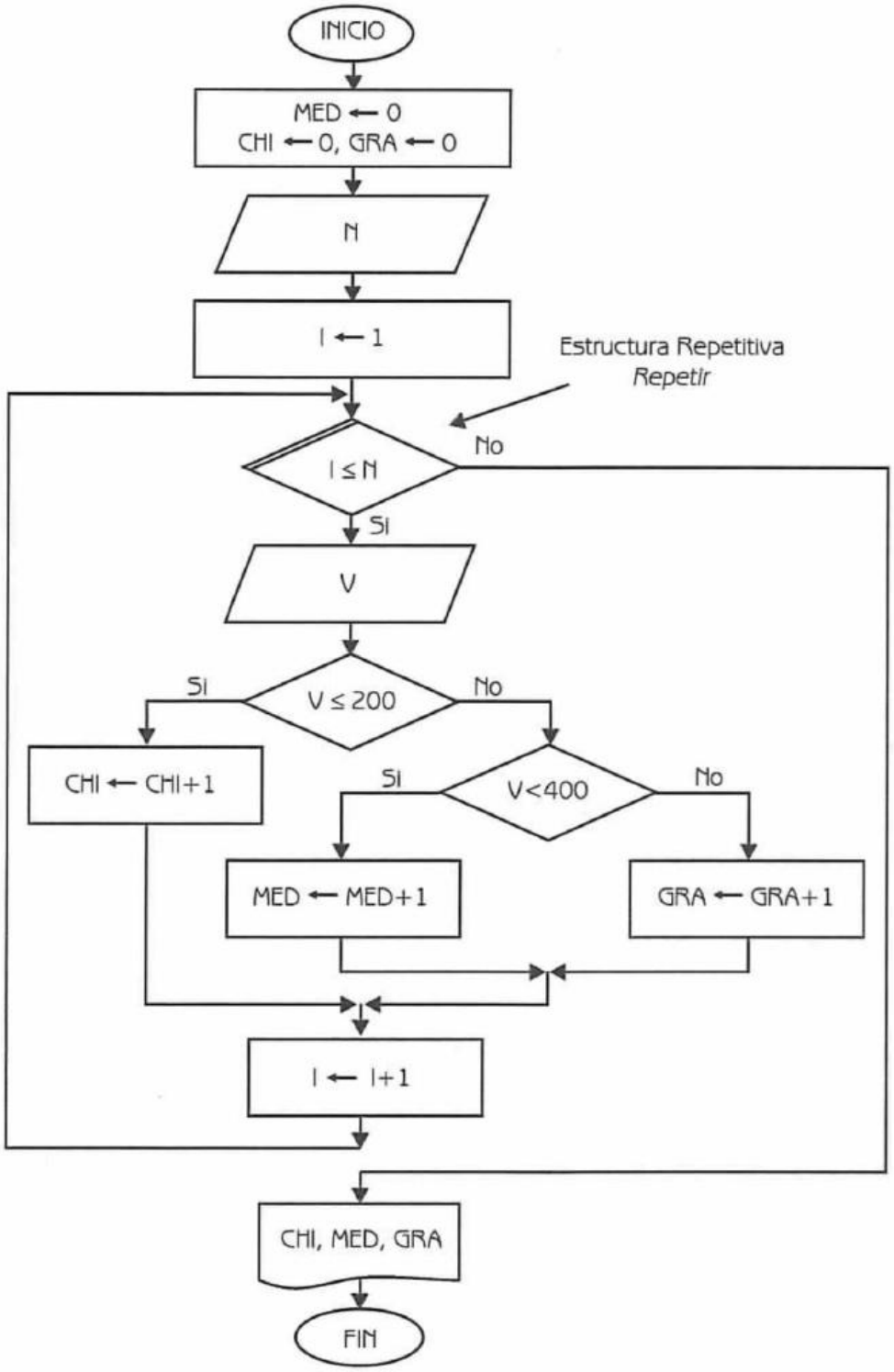


Diagrama de Flujo 3.14

N: Variable de tipo entero.
V: Variable de tipo real.

A continuación en la tabla 3.9, podemos observar el seguimiento del algoritmo.

www.FreeLibros.com

Tabla 3.9

I	N	MED	CHI	GRA	V
1	12	0	0	0	
2			1		180.25
3				1	470.30
4			2		150.25
5			3		88.60
6		1			230.15
7			4		170.20
8			5		40.30
9		2			201.90
10			6		60.32
11		3			280.30
12			7		15.70
13			8		140.20

Inicio del Ciclo ⟶

Fin del Ciclo ⟶

: Expresa valores que se imprimen.

Programa 3.13

```
VENDEDOR

{El programa dada información sobre las ventas que hizo un vendedor, obtiene
cuantas ventas fueron chicas, medianas y grandes}

{I, CHI, MED, GRA y N son variables de tipo entero. V es una variable de tipo real}

1. Hacer CHI ⟵ 0, MED ⟵ 0 y GRA ⟵ 0
2. Leer N
3. Hacer I ⟵ 1
4. Repetir con I desde 1 hasta N
        Leer V
    4.1 Si V <= 200
            entonces
                    Hacer CHI ⟵ CHI + 1
            sino
            4.1.1 Si V < 400
                        entonces
                                Hacer MED ⟵ MED + 1
                        sino
                                Hacer GRA ⟵ GRA + 1
            4.1.2 {Fin del condicional del paso 4.1.1}
    4.2 {Fin del condicional del paso 4.1}
        Hacer I ⟵ I + 1
5. {Fin del ciclo del paso 4}
6. Escribir CHI, MED y GRA
```

www.FreeLibros.com

Problema 3.8

La siguiente se llama la conjetura de ULAM en honor del matemático S.Ulam:

- Comience con cualquier entero positivo
- Si es par, divídalo entre 2; si es impar, multiplíquelo por 3 y agréguele 1.
- Obtenga enteros sucesivamente repitiendo el proceso.

Al final, obtendrá el número 1, independientemente del entero inicial. Por ejemplo, cuando el entero inicial es 26, la secuencia será: 26, 13, 40, 20, 10, 5, 16, 8, 4, 2, 1.

Construya un diagrama de flujo que lea un entero positivo y obtenga e imprima la sucesión de ULAM.

Dato: NUM (variable de tipo entero que representa el número que se ingresa).

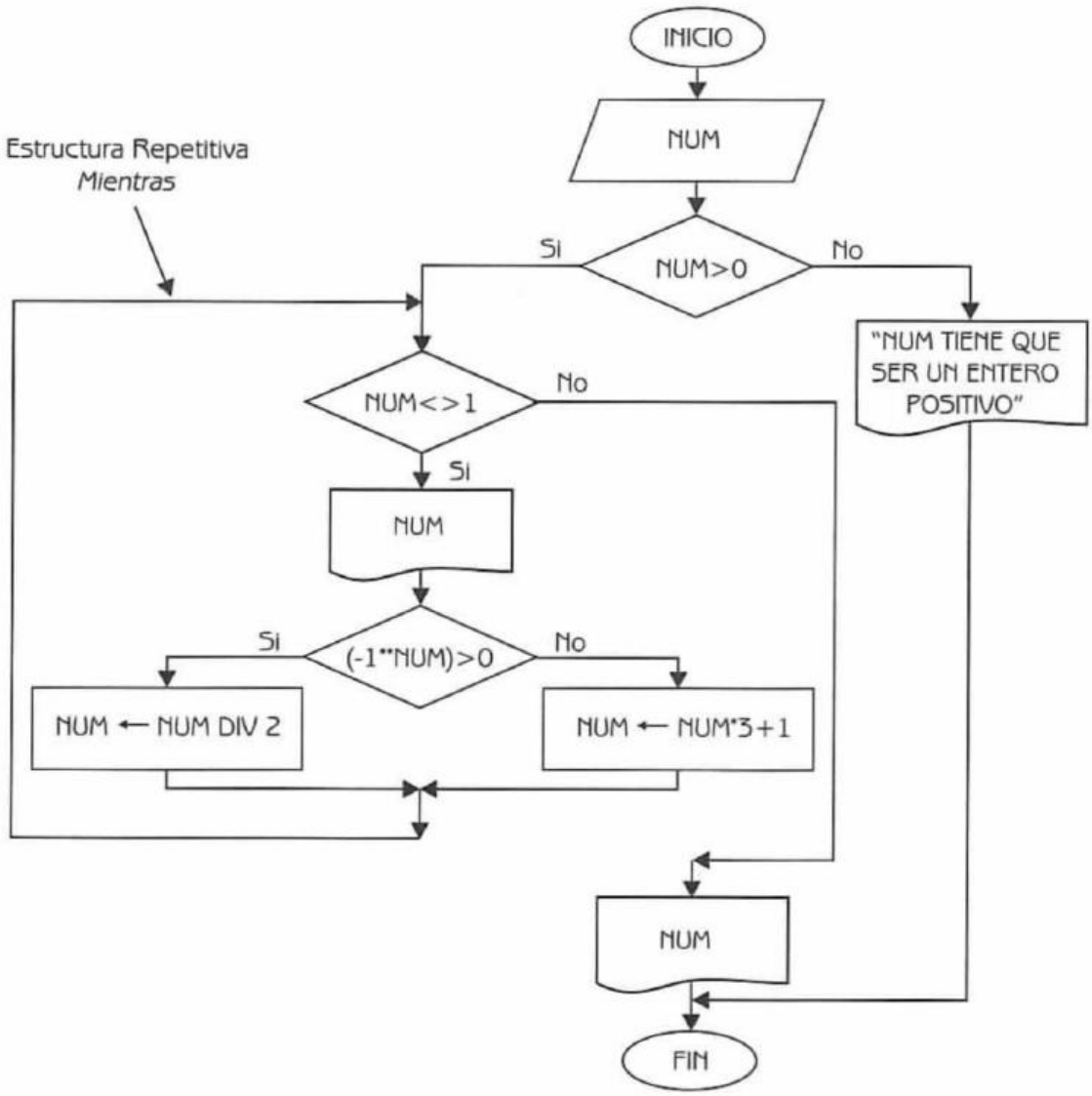


Diagrama de Flujo 3.15

www.FreeLibros.com

Explicación de las variables

NUM: Variable de tipo entero.

A continuación en la siguiente tabla presentamos el seguimiento del algoritmo.

Tabla 3.10

NUM	
25	
76	Inicio del Ciclo →
38	
19	
58	
29	
88	
44	
22	
11	
34	
17	
52	
26	
13	
40	
20	
10	
5	
16	
8	
4	
2	
1	Fin del Ciclo →

: Expresa valores que se imprimen.

www.FreeLibros.com

Programa 3.14

```
ULAM

{El programa, dado un entero positivo, obtiene e imprime la sucesión de ULAM}

{NUM es una variable de tipo entero}

1. Leer NUM
2. Si NUM > 0
       entonces
         2.1 Mientras (NUM < > 1) Repetir
              Escribir NUM
            2.1.1 Si (-1 ** NUM) > 0
                              entonces
                                   Hacer NUM ←—— NUM DIV 2
                              sino
                                   Hacer NUM ←—— NUM * 3 + 1
            2.1.2 {Fin del condicional del paso 2.1.1}
         2.2 {Fin del ciclo del paso 2.1}
              Escribir NUM
         sino
              Escribir " NUM tiene que ser un entero positivo"
3. {Fin del condicional del paso 2}
```

Problema 3.9

Escriba un diagrama de flujo que lea un número entero N y calcule la suma de la siguiente serie:

$$1^1 + 2^2 + 3^3 + \ldots N^N$$

Dato: N (variable de tipo entero que representa el número de términos de la serie).

www.FreeLibros.com

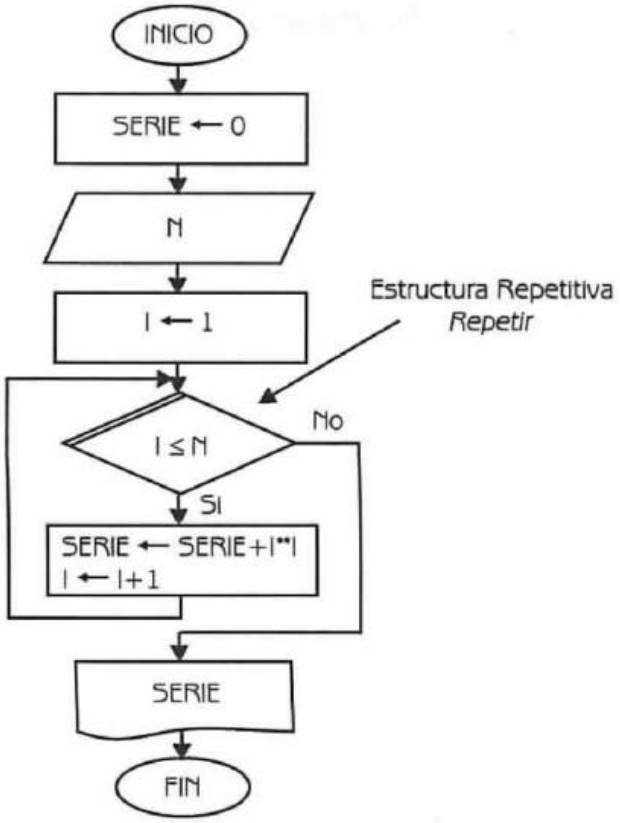


Diagrama de Flujo 3.16

Explicación de las variables

I: Variable de tipo entero. Representa la variable de control del ciclo.
N: Variable de tipo entero.
SERIE: Variable de tipo real. Es un acumulador. Acumula los resultados de cada término, obteniendo finalmente el resultado de la serie.

A continuación en la tabla 3.11 podemos observar el seguimiento del algoritmo.

Tabla 3.11

I	N	SERIE
1	8	0
2		1
3		5
4		32
5		288
6		3413
7		50069
8		873612
9		17650828

Inicio del Ciclo →

Fin del Ciclo →

: Expresa el valor que se imprime.

www.FreeLibros.com

Programa 3.15

```
SERIE
{El programa calcula la suma de una serie}
{I y N son variables de tipo entero. Serie es una variable de tipo real}
1. Hacer SERIE ⟵ 0
2. Leer N
3. Hacer I ⟵ 1
4. Repetir con I desde 1 hasta N
        Hacer SERIE ⟵ SERIE + I ** I e I ⟵ I + 1
5. {Fin del ciclo del paso 4}
6. Escribir SERIE
```

Problema 3.10

Haga un diagrama de flujo que calcule el término número 180 de la secuencia FIBONACCI. Recuerde que los dos primeros números de la serie son 0 y 1. El resto se calcula como la suma de los dos números inmediatos que le preceden.

Ejemplo de la serie: 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, . . . ,

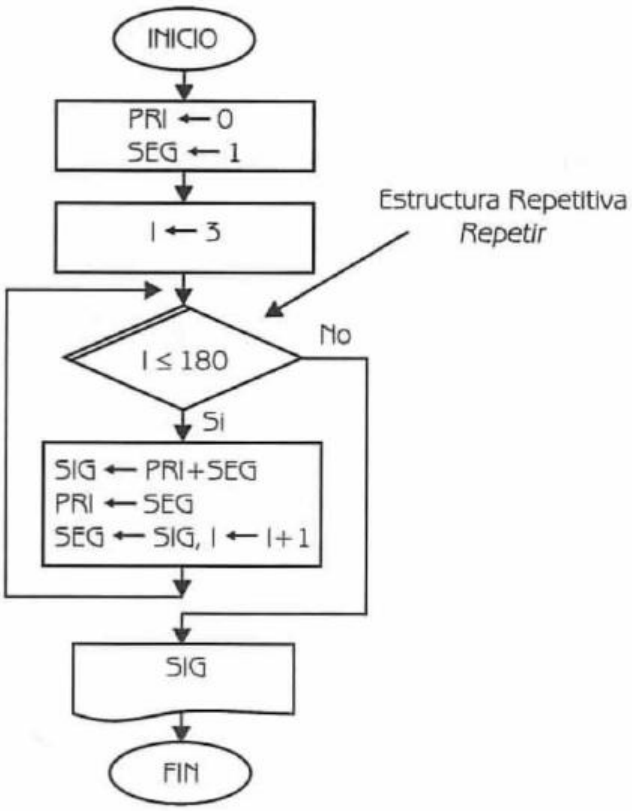


Diagrama de Flujo 3.17

www.FreeLibros.com

Explicación de las variables

I: Variable de tipo entero. Representa la variable de control del ciclo. Normalmente se inicializa en 1, en este caso se inicializa en 3 puesto que existen dos asignaciones previas al comienzo del ciclo.

PRI: Variable de tipo entero. Representa al primero de los dos números que hay que sumar para obtener el siguiente número de la serie.

SEG : Variable de tipo entero. Representa al segundo de los dos números que hay que sumar para obtener el siguiente número de la serie.

SIG : Variable de tipo entero. Representa el siguiente número de la serie y se obtiene como la suma de (PRI + SEG)

En la siguiente tabla podemos observar el seguimiento del algoritmo. Consideremos además que en lugar de imprimir el término 680, calcularemos el término número 7.

Tabla 3.12

	I	PRI	SEG	SIG
	3	0	1	
Inicio del Ciclo →	4	1	1	1
	5	1	2	2
	6	2	3	3
	7	3	5	5
Fin del Ciclo →	8	5	8	(8)

(8) : Expresa el valor que se imprime.

www.FreeLibros.com

Programa 3.16

```
FIBONACCI_680

{El programa calcula el término 180 de la secuencia FIBONACCI}

{I, PRI, SEG y SIG son variables de tipo entero}

1. Hacer PRI ←— 0, SEG ←— 1, I ←— 3
2. Repetir con I desde 3 hasta 180
      Hacer SIG ←— PRI + SEG, PRI ←— SEG, SEG ←— SIG e I ←— I + 1
3. {Fin del ciclo del paso 2}
4. Escribir SIG
```

Problema 3.11

Supóngase que en una reciente elección hubo cuatro candidatos (con identificadores 1, 2, 3, 4). Usted habrá de encontrar, mediante un programa, el número de votos correspondiente a cada candidato y el porcentaje que obtuvo respecto al total de los votantes. El usuario tecleará los votos de manera desorganizada, tal y como se obtuvieron en la elección, el final de datos está representado por un cero. Observe, como ejemplo, la siguiente lista:

1 3 1 4 2 2 1 4 1 1 1 2 1 3 1 4 0

Donde 1 representa un voto para el candidato 1; 3 un voto para el candidato 3; y así sucesivamente.

Datos: $VOTO_1, VOTO_2, \ldots, 0$

Donde:

$VOTO_i$ es una variable de tipo entero que representa el voto para uno de los cuatro candidatos.

www.FreeLibros.com

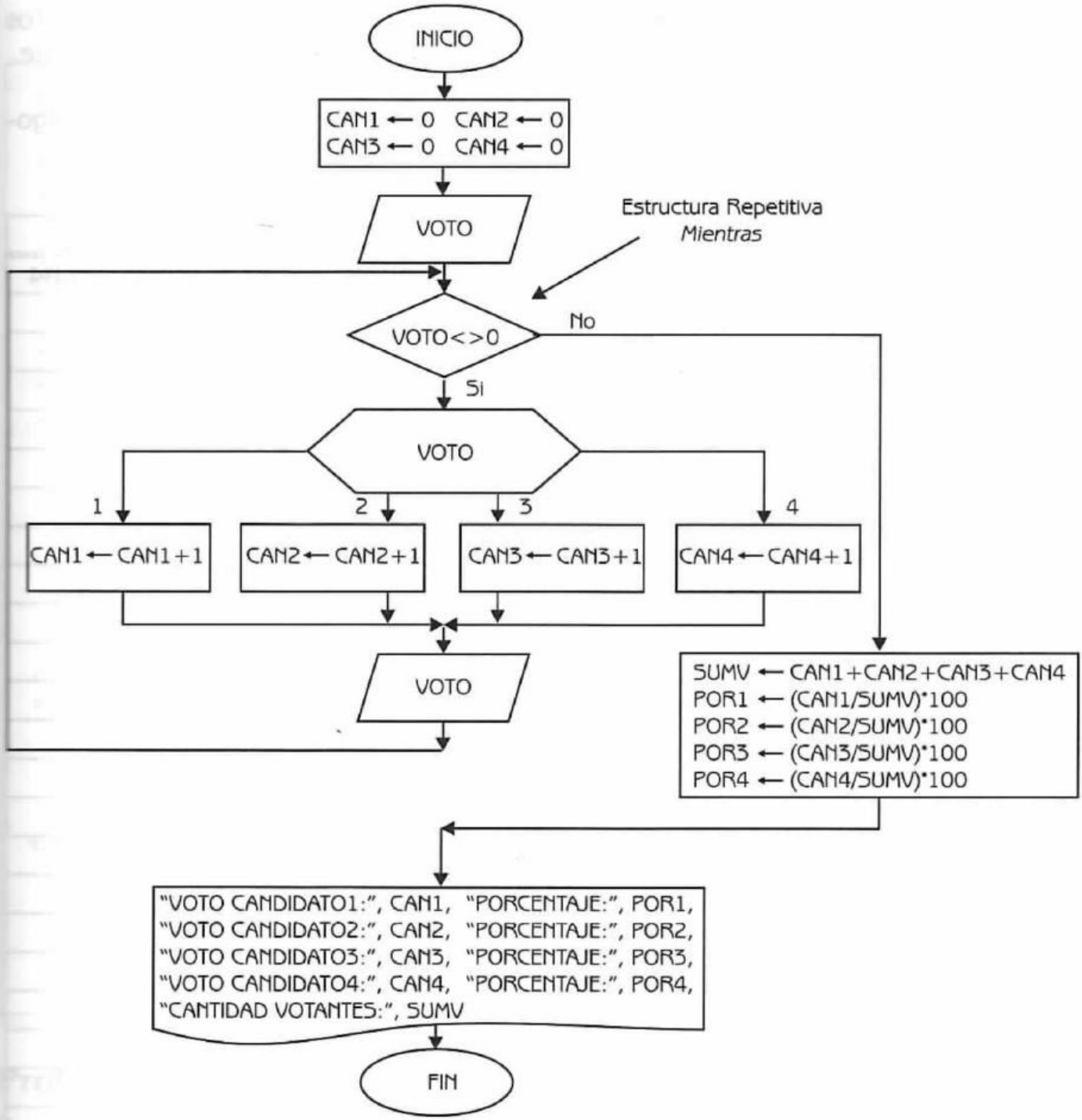


Diagrama de Flujo 3.18

Explicación de las variables

CAN1, *CAN2*, *CAN3*, *CAN4*: Variables de tipo real. Son acumuladores. Acumulan el número de votos de los candidatos 1, 2, 3, y 4, respectivamente.

VOTO: Variable de tipo entero.

SUMV: Variable de tipo real. Almacena el total de votos emitidos en la elección.

www.FreeLibros.com

POR1, *POR2*, *POR3*, *POR4*: Variables de tipo real. Almacenan el porcentaje de votos obtenidos por los candidatos 1, 2, 3 y 4, respectivamente.

A continuación en la siguiente tabla podemos observar el seguimiento del algoritmo.

Tabla 3.13

VOTO	CAN1	CAN2	CAN3	CAN4	SUMV	POR1	POR2	POR3	POR4
1	0	0	0	0					
1	1								
2	2								
1		1							
3	3								
1			1						
1	4								
2	5								
4		2							
4				1					
3				2					
1			2						
1	6								
2	7								
3		3							
1			3						
4	8								
1				3					
2	9								
1		4							
2	10								
4		5							
4				4					
2				5					
1		6							
0	11								
					25	44.00	24.00	12.00	20.00

: Expresa valores que se imprimen.

www.FreeLibros.com

Programa 3.17

```
ELECCIONES

{El programa, obtiene el total de votos y el porcentaje obtenido por los cuatro candidatos de una elección}

{VOTO es una variable de tipo entero. POR1, POR2, POR3, POR4, CAN1, CAN2, CAN3, CAN4, y SUMV son variables de tipo real}

1. Hacer CAN1 ←— 0, CAN2 ←— 0, CAN3 ←— 0 y CAN4 ←— 0
2. Leer VOTO
3. Mientras (VOTO < > 0) Repetir
     3.1 Si VOTO Igual
           1 : Hacer CAN1 ←— CAN1 + 1
           2 : Hacer CAN2 ←— CAN2 + 1
           3 : Hacer CAN3 ←— CAN3 + 1
           4 : Hacer CAN4 ←— CAN4 + 1
     3.2 {Fin del condicional del paso 3.1}
         Leer VOTO
4. {Fin del ciclo del paso 3}
5. Hacer SUMV ←— CAN1 + CAN2 + CAN3 + CAN4
       POR1 ←— (CAN1/SUMV) * 100
       POR2 ←— (CAN2/SUMV) * 100, POR3 ←— (CAN3/SUMV) * 100 y
       POR4 ←— (CAN4/SUMV) * 100
6. Escribir " VOTOS CANDIDATO 1 : ", CAN1, " PORCENTAJE : ", POR1
            " VOTOS CANDIDATO 2 : ", CAN2, " PORCENTAJE : ", POR2
            " VOTOS CANDIDATO 3 : ", CAN3, " PORCENTAJE : ", POR3
            " VOTOS CANDIDATO 4 : ", CAN4, " PORCENTAJE : ", POR4
            " CANTIDAD VOTANTES : ", SUMV
```

Problema 3.12

De una empresa con N empleados, necesitamos obtener el número de empleado y sueldo del trabajador con el mayor sueldo de la empresa. Haga el diagrama de flujo correspondiente.

Datos: N, $NUMEMP_1$, SUE_1, $NUMEMP_2$, SUE_2, . . ., $NUMEMP_N$, SUE_N

Donde:

N es una variable de tipo entero que representa el número de empleados de la empresa.

www.FreeLibros.com

$NUMEMP_i$ es una variable de tipo entero que representa el número del empleado i ($1 \leq i \leq N$).

SUE_i es una variable de tipo real que representa el sueldo del empleado i ($1 \leq i \leq N$).

Nota sobre la variable MASUE:

Cuando se trata de localizar un máximo, la variable se inicializa con un valor mínimo. En la primera vuelta del ciclo, *MASUE* toma el valor del primer sueldo.

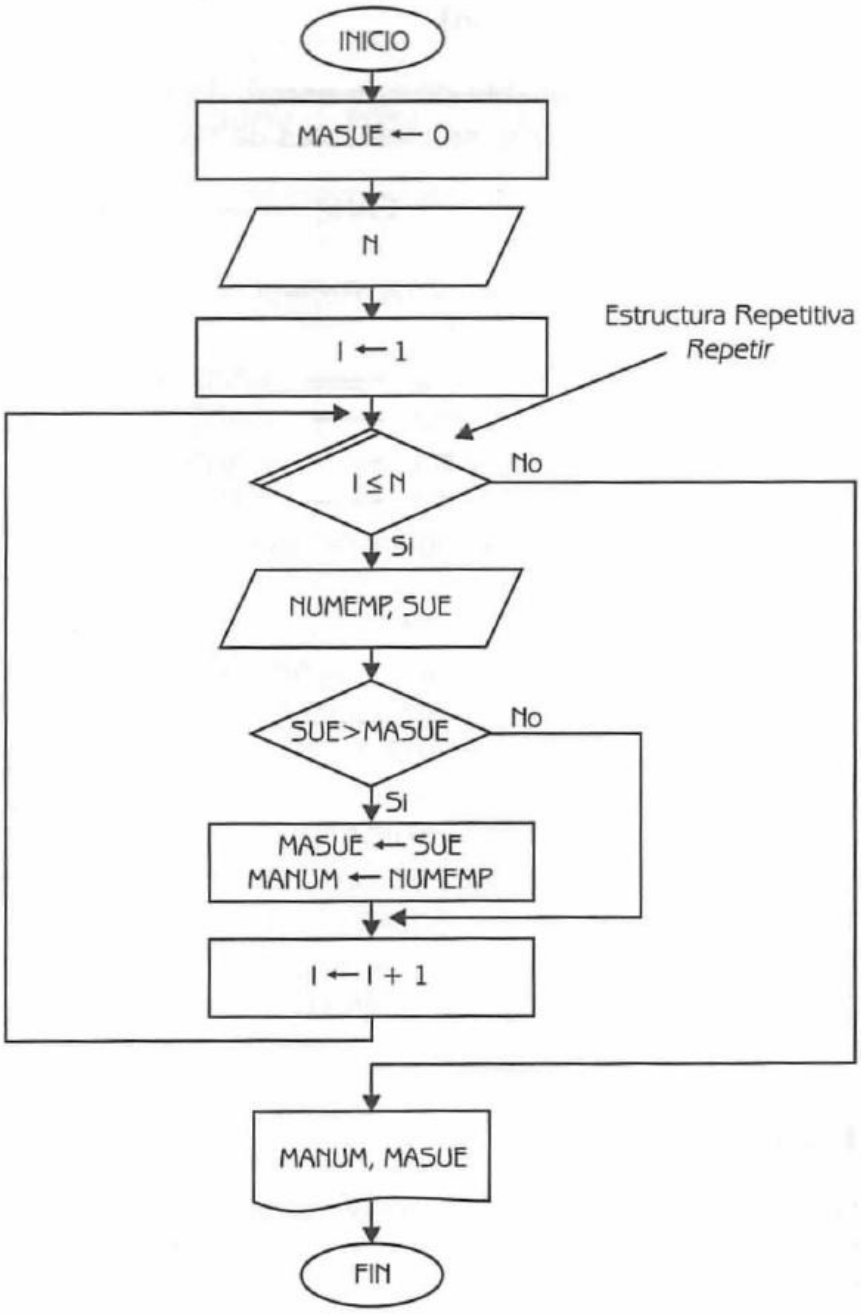


Diagrama de Flujo 3.19

Explicación de las variables

I: Variable de tipo entero. Representa la variable de control del ciclo.

www.FreeLibros.com

N: Variable de tipo entero.

NUMEMP: Variable de tipo entero.

SUE: Variable de tipo real.

MASUE: Variable de tipo real. Almacena el mayor sueldo de los empleados.

MANUM: Variable de tipo entero. Almacena el número del empleado que tenga el mayor sueldo.

En la tabla 3.14 podemos seguir el algoritmo para un valor de N = 6. Los datos que ingresaremos son los siguientes:

NUMEMP	*SUE*
1235	2580
1260	1400
1325	3600
1418	2720
1650	4120
1717	1950

Tabla 3.14

	I	N	NUMEMP	SUE	MASUE	MANUM
	1	6			0	
Inicio del Ciclo →	2		1235	2580	2580	1235
	3		1260	1400		
	4		1325	3600	3600	1325
	5		1418	2720		
	6		1650	4120	4120	1650
Fin del Ciclo →	7		1717	1950		

: Señala los valores que se imprimen.

www.FreeLibros.com

Programa 3.18

```
MAYOR_SUELDO

{El programa obtiene el número de empleado y el sueldo, del trabajador con
mayor ingreso}

{I, N, NUMEMP y MANUM son variables de tipo entero. SUE y MASUE son variables
de tipo real}

1. Hacer MASUE ←— 0
2. Leer N
3. Hacer I ←— 1
4. Repetir con I desde 1 hasta N
        Leer NUMEMP y SUE
   4.1 Si SUE > MASUE entonces
            Hacer MASUE ←— SUE y MANUM ←— NUMEMP
   4.2 {Fin del condicional del paso 4.2}
       Hacer I ←— I + 1
5. {Fin del ciclo del paso 4}
6. Escribir MANUM y MASUE
```

Problema 3.13

En el centro meteorológico de Argentina se llevan los promedios mensuales de las lluvias caídas en las principales regiones cerealeras del país. Existen 3 regiones importantes denominadas NORTE, CENTRO y SUR. Haga un diagrama de flujo para calcular lo siguiente:

a) El promedio anual de la región centro.
b) El mes y registro con menor lluvia en la región sur.
c) La región con mayor lluvia anual (Considere que los registros anuales de las regiones son diferentes).

Datos: $RNO_1, RCE_1, RSU_1, \ldots, RNO_{12}, RCE_{12}, RSU_{12}$

Donde:

RNO_i es una variable de tipo real que indica la lluvia caída en la región norte en el mes i ($1 \leq i \leq 12$).
RCE_i es una variable de tipo real que expresa la lluvia caída en la región centro en el mes i ($1 \leq i \leq 12$).
RSU_i es una variable de tipo real que representa la lluvia caída en la región sur en el mes i ($1 \leq i \leq 12$).

www.FreeLibros.com

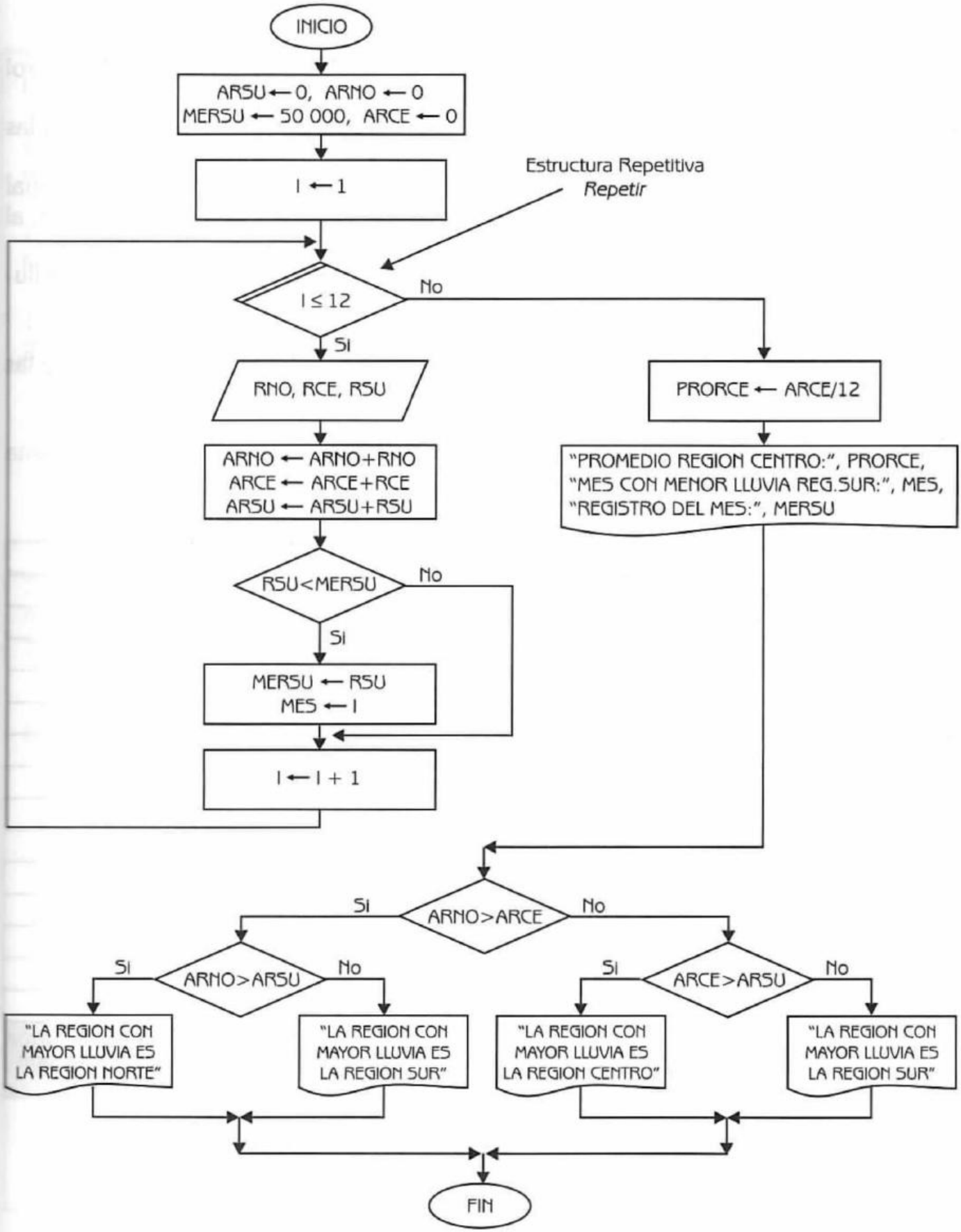


Diagrama de Flujo 3.20

www.FreeLibros.com

Explicación de las variables

I: Variable de tipo entero. Representa la variable de control del ciclo.
ARNO, *ARCE* y *ARSU*: Variables de tipo real. Acumulan las lluvias caídas en las regiones norte, centro y sur.
MERSU: Variable de tipo real. Almacena el menor registro mensual de la región sur. Como se trata de localizar un mínimo, al principio se inicializa con un valor muy elevado.
MES: Variable de tipo entero. Almacena el mes con menores lluvias en la región sur.
RNO, *RCE* y *RSU*: Variables de tipo real.
PRORCE: Variable de tipo real. Almacena el promedio anual de las lluvias caídas en la región centro.

A continuación en la tabla 3.15 podemos observar el seguimiento del diagrama de flujo.

Tabla 3.15

I	RNO	RCE	RSU	ARNO	ARCE	ARSU	MERSU	MES	PRORCE	
1				0	0	0	500 000			
2	80	70	170	80	70	170	170	1		
3	60	85	100	140	155	270	100	2		
4	120	100	185	260	255	455				
5	100	90	88	360	345	543	88	4		
6	70	82	80	430	427	623	80	5		
7	150	130	122	580	557	745				
8	100	95	67	680	652	812	67	7		
9	47	38	55	727	690	867	55	8		
10	95	115	110	822	805	977				
11	70	62	83	892	867	1060				
12	100	87	107	992	954	1167				
13	130	103	148	1122	1057	1315				
									88.08	"LA REGION CON MAYOR LLUVIA ES LA REGION SUR"

: Expresa valores que se imprimen

www.FreeLibros.com

Programa 3.19

```
LLUVIAS

{El programa dada información sobre las lluvias caídas en las principales zonas cerealeras de Argentina, realiza cálculos estadísticos}

{I y MES son variables de tipo entero. ARNO, ARCE, ARSU, MERSU, RNO, RCE, RSU y PRORCE son variables de tipo real}

1. Hacer ARNO ←— 0, ARCE ←— 0, ARSU ←— 0, MERSU ←— 500 000 e I ←— 1
2. Repetir con I desde 1 hasta 12
        Leer RNO, RCE y RSU
        Hacer ARNO ←— ARNO + RNO, ARCE ←— ARCE + RCE y
                ARSU ←— ARSU + RSU
    2.1 Si RSU < MERSU entonces
                Hacer MERSU ←— RSU y MES ←— I
    2.2. {Fin del condicional del paso 2.1}
          Hacer I ←— I + 1
3. {Fin del ciclo del paso 2}
4. Hacer PRORCE ←— ARCE / 12
5. Escribir " PROMEDIO REGION : ", PRORCE,
             " MES CON MENOR LLUVIA REGION SUR : ", MES,
             " REGISTRO DEL MES : ", MERSU
6. Si ARNO > ARCE
        entonces
        6.1 Si ARNO > ARSU
                entonces
                        Escribir " LA REGION CON MAYOR LLUVIA ES LA NORTE "
                sino
                        Escribir " LA REGION CON MAYOR LLUVIA ES LA SUR "
        6.2 {Fin del condicional del paso 6.1}
        sino
        6.3 Si ARCE > ARSU
                entonces
                        Escribir " LA REGION CON MAYOR LLUVIA ES LA CENTRO"
                sino
                        Escribir " LA REGION CON MAYOR LLUVIA ES LA SUR "
        6.4 {Fin del condicional del paso 6.3}
7. {Fin del condicional del paso 6}
```

www.FreeLibros.com

Problema 3.14

En un estadio se tienen 5 tipos diferentes de localidades, las cuales se identifican por una clave numérica que es un valor comprendido entre 1 y 5. Los precios de cada localidad y los datos referentes a las ventas de boletos para el próximo juego se proporcionan como sigue:

Datos: P1, P2, P3, P4, P5

$CLAVE_1$, $CANT_1$
$CLAVE_2$, $CANT_2$
. . . . . .
−1 , −1

Donde :

P1, P2, P3,P4 y P5	son variables de tipo real que representan los precios de las localidades 1, 2, 3, 4 y 5, respectivamente.
$CLAVE_i$	es una variable de tipo entero que representa el tipo de localidad de la venta i.
$CANT_i$	es una variable de tipo entero que indica la cantidad de boletos vendidos de un cierto tipo, en la venta i.

Construya un diagrama de flujo que:

a) Lea los precios.

b) Lea los datos de las ventas de boletos.

c) Imprima para cada venta, la clave, la cantidad y el importe total de los boletos vendidos en esta venta.

d) Calcule e imprima la cantidad de boletos vendidos de cada tipo.

e) Calcule e imprima la recaudación total del estadio.

Nota Considere que en una venta sólo se venden boletos de un tipo.

www.FreeLibros.com

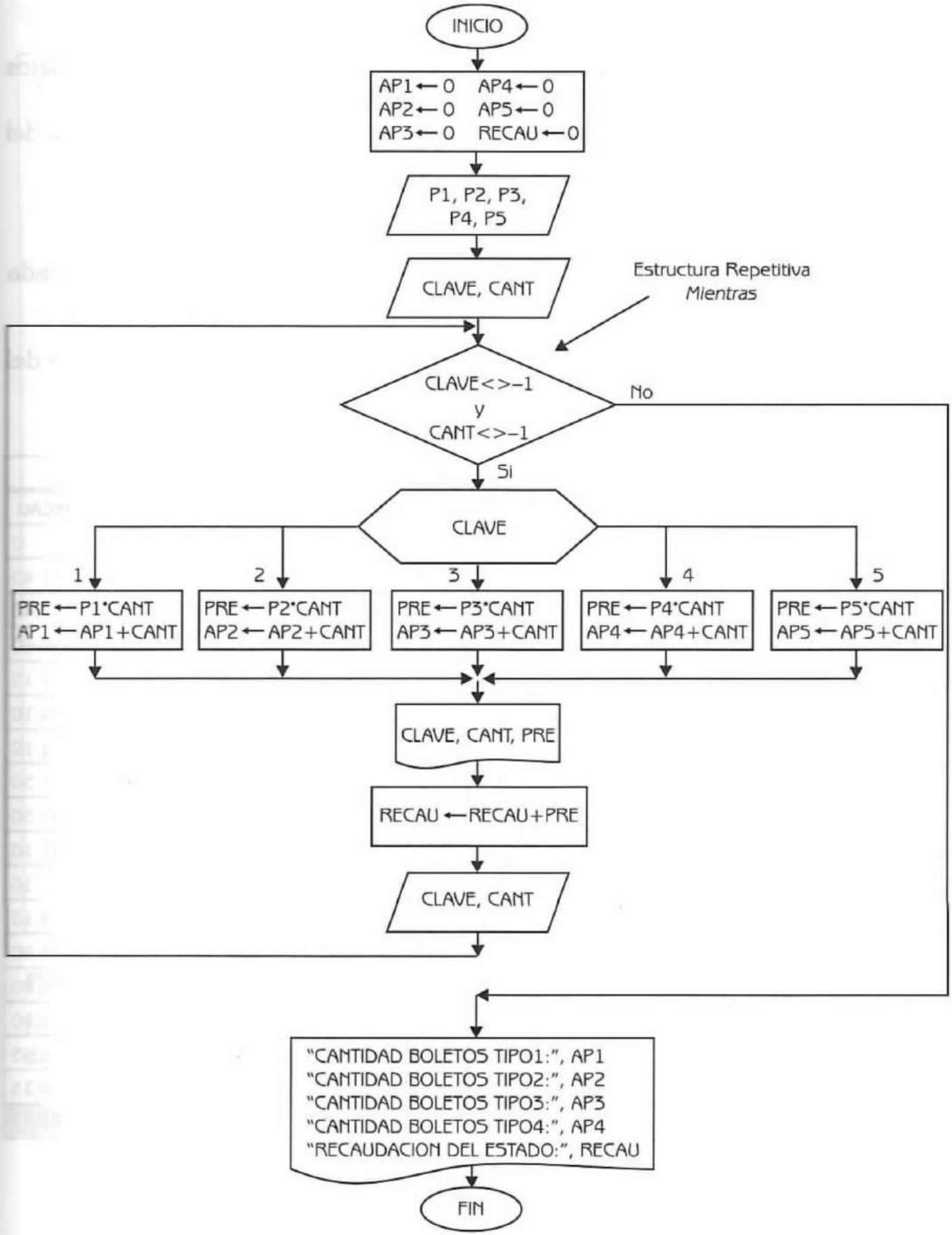


Diagrama de Flujo 3.21

www.FreeLibros.com

Explicación de las variables

AP1, AP2, AP3, AP4 y AP5: Variables de tipo entero. Acumulan el total de boletos vendidos del tipo 1, 2, 3, 4 y 5, respectivamente.

RECAU: Variable de tipo real. Acumula la recaudación total del estadio.

P1, P2, P3, P4 y P5: Variables de tipo real.

CLAVE y CANT: Variable de tipo entero.

PRE: Variable de tipo real. Almacena el total vendido en cada venta.

A continuación en la siguiente tabla, podemos observar el seguimiento del diagrama de flujo.

Tabla 3.16

	P1	P2	P3	P4	P5	CLAVE	CANT	PRE	AP1	AP2	AP3	AP4	AP5	RECAU
Inicia el Ciclo →	7.25	15.80	25.00	50.00	75.00	2	3		0	0	0	0	0	0
						3	8	47.40		3				47.40
						2	4	200.00			8			247.40
						1	6	63.20		7				310.60
						4	5	43.5	6					354.10
						1	12	250.00				5		604.10
						2	8	87.00	18					691.10
						5	3	126.40		15				817.50
						2	7	225.00					3	1042.50
						3	14	110.60		22				1153.10
						4	1	350.00			22			1503.10
						2	11	50.00				6		1553.10
						4	9	173.80		33				1726.90
						5	7	450.00				15		2176.90
						1	23	525.00					10	2701.90
						1	18	166.75	41					2868.65
						3	4	130.50	59					2999.15
Fin del Ciclo →						-1	-1	100.00			26			3099.15

: Expresa cantidades que se imprimen

www.FreeLibros.com

Programa 3.20

```
ESTADIO_PRECIOS

{El programa, dado los precios de distintas localidades y las cantidades vendidas de
cada una de ellas en una determinada venta, obtiene el total vendido en esa venta, las
cantidades de boletos vendidos de cada localidad y la recaudación total del estadio}

{AP1, AP2, AP3, AP4, AP5, CLAVE y CANT son variables de tipo entero. RECAU, P1,
P2, P3, P4, P5 y PRE son variables de tipo real}

1. Hacer AP1 ⟵ 0, AP2 ⟵ 0, AP3 ⟵ 0, AP4 ⟵ 0, AP5 ⟵ 0, y
     RECAU ⟵ 0
2. Leer P1, P2, P3, P4 y P5
3. Leer CLAVE y CANT
4. Mientras (CLAVE < > – 1) y (CANT < > –1) Repetir
        4.1 Si CLAVE Igual
              1 : Hacer PRE ⟵ P1 * CANT y AP1 ⟵ AP1 + CANT
              2 : Hacer PRE ⟵ P2 * CANT y AP2 ⟵ AP2 + CANT
              3 : Hacer PRE ⟵ P3 * CANT y AP3 ⟵ AP3 + CANT
              4 : Hacer PRE ⟵ P4 * CANT y AP4 ⟵ AP4 + CANT
              5 : Hacer PRE ⟵ P5 * CANT y AP5 ⟵ AP5 + CANT
        4.2 {Fin del condicional del paso 4.1}
              Escribir CLAVE, CANT y PRE
              Hacer RECAU ⟵ RECAU + PRE
              Leer CLAVE y CANT
5. {Fin del ciclo del paso 4}
6. Escribir " CANTIDAD BOLETOS TIPO1 : ", AP1,
            " CANTIDAD BOLETOS TIPO2 : ", AP2,
            " CANTIDAD BOLETOS TIPO3 : ", AP3,
            " CANTIDAD BOLETOS TIPO4 : ", AP4,
            " CANTIDAD BOLETOS TIPO5 : ", AP5,
            " RECAUDACION DEL ESTADIO : ", RECAU
```

Problema 3.15

Haga un diagrama de flujo para calcular lo que hay que pagar por un conjunto de llamadas telefónicas. Por cada llamada se ingresa el tipo (Internacional, Nacional, Local) y la duración en minutos. El criterio que se sigue para calcular el costo de cada llamada es el siguiente:

Internacional:	3 primeros minutos $7.59
	Cada minuto adicional $3.03
Nacional:	3 primeros minutos $1.20
	Cada minuto adicional $0.48

www.FreeLibros.com

Local: Las primeras 50 llamadas no se cobran. Luego, cada llamada cuesta $0.60

Datos: $TIPO_1$, DUR_1, $TIPO_2$, DUR_2,...,X,–1

Donde:

$TIPO_i$ es una variable de tipo caracter que expresa el tipo de la llamada i. Toma el valor de 'I' si la llamada es internacional, 'N' si es nacional y 'L' si es local.

DUR_i es una variable de tipo entero que indica la duración de la llamada i en minutos.

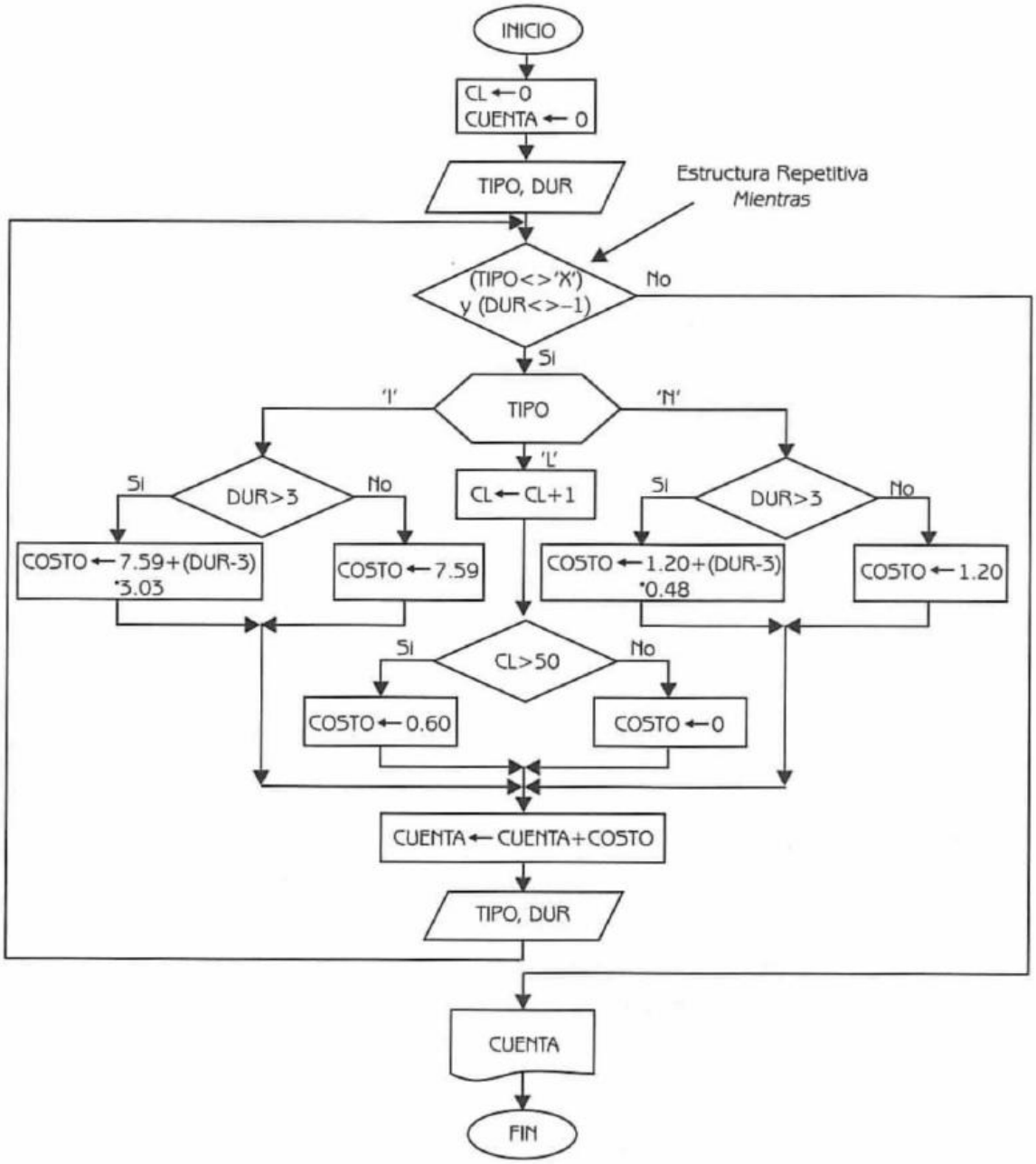


Diagrama de Flujo 3.22

www.FreeLibros.com

Explicación de las Variables

CL: Variable de tipo entero. Acumula el número de llamadas locales.

CUENTA: Variable de tipo real. Acumula el costo de cada llamada.

TIPO: Variable de tipo caracter.

DUR: Variable de tipo entero.

COSTO: Variable de tipo real. Almacena el costo de cada llamada.

En la siguiente tabla se puede observar el seguimiento del algoritmo para el siguiente conjunto de llamadas.

Datos: I, 7, N, 6, N, 12 ,L ,5 ,L ,7 ,L ,15, N, 16, L, 7, L, 6, L, 4, I, 11, X, –1

Tabla 3.17

	TIPO	DUR	CL	COSTO	CUENTA
	I	7	0		0
Inicia el Ciclo →	N	6		19.71	19.71
	N	12		2.64	22.35
	L	5		5.52	27.87
	L	7	1	0	27.87
	L	15	2	0	27.87
	N	16	3	0	27.87
	L	7		7.44	35.31
	L	6	4	0	35.31
	L	4	5	0	35.31
	I	11	6	0	35.31
Fin del Ciclo →	X	–1		31.83	67.14

: Expresa el valor que se imprime.

www.FreeLibros.com

Programa 3.21

```
TELEFONO

{El programa calcula el costo de un conjunto de llamadas telefónicas}

{CL y DUR son variables de tipo entero. CUENTA y COSTO son variables de tipo real.
TIPO es una variable de tipo caracter}

1. Hacer CL ←— 0  y  CUENTA ←— 0
2. Leer TIPO y DUR
3. Mientras (TIPO < > "X")  y  (DUR < > –1) Repetir
     3.1 Si TIPO Igual
           "I":
           3.1.1 Si DUR > 3
                       entonces
                            Hacer COSTO ←— 7.59 + ((DUR–3) *3.03)
                       sino
                            Hacer COSTO ←— 7.59
           3.1.2 {Fin del condicional 3.1.1.}
           " L " : Hacer CL ←— CL+ 1
           3.1.3 Si CL > 50
                       entonces
                            Hacer COSTO ←— 0.60
                       sino
                            Hacer COSTO ←— 0
           3.1.4 {Fin del condicional 3.13}
           " N " :
           3.1.5 Si DUR > 3
                       entonces
                            Hacer COSTO ←— 1.20 + ((DUR-3)*0.48)
                       sino
                            Hacer COSTO ←— 1.20
           3.1.6 {Fin del condicional 3.1.5}
     3.2 {Fin del condicional del paso 3.1}
           Hacer CUENTA ←— CUENTA + COSTO
           Leer TIPO y DUR
4. {Fin del ciclo del paso 3}
5. Escribir CUENTA
```

www.FreeLibros.com

Problema 3.16

En una bodega se tiene información sobre las cantidades producidas de cada tipo de vino, a lo largo de los últimos años. Haga un diagrama de flujo que calcule e imprima lo siguiente:

a) El total producido de cada tipo de vino (son 5 tipos) a lo largo de los N años.

b) El total producido de vino por año.

c) Año en que se produjo la mayor cantidad de litros de vino del tipo 2. Imprimir también la cantidad de litros.

d) Verificar si hubo algún año en el cual no se produjo el vino tipo 3. Si existe dicho año, imprimirlo.

Datos: N

$V_{1,1}$, $V_{1,2}$,...,$V_{1,5}$
$V_{2,1}$, $V_{2,2}$,...,$V_{2,5}$
. . .
$V_{N,1}$, $V_{N,2}$,...,$V_{N,5}$

Donde:

N es una variable de tipo entero que representa el número de años, sobre los que se obtendrá la información pedida.

$V_{i,j}$ es una variable de tipo real que indica la cantidad de litros de vino del tipo j, producidos en el año i ($1 \leq i \leq N$, $1 \leq j \leq 5$).

www.FreeLibros.com

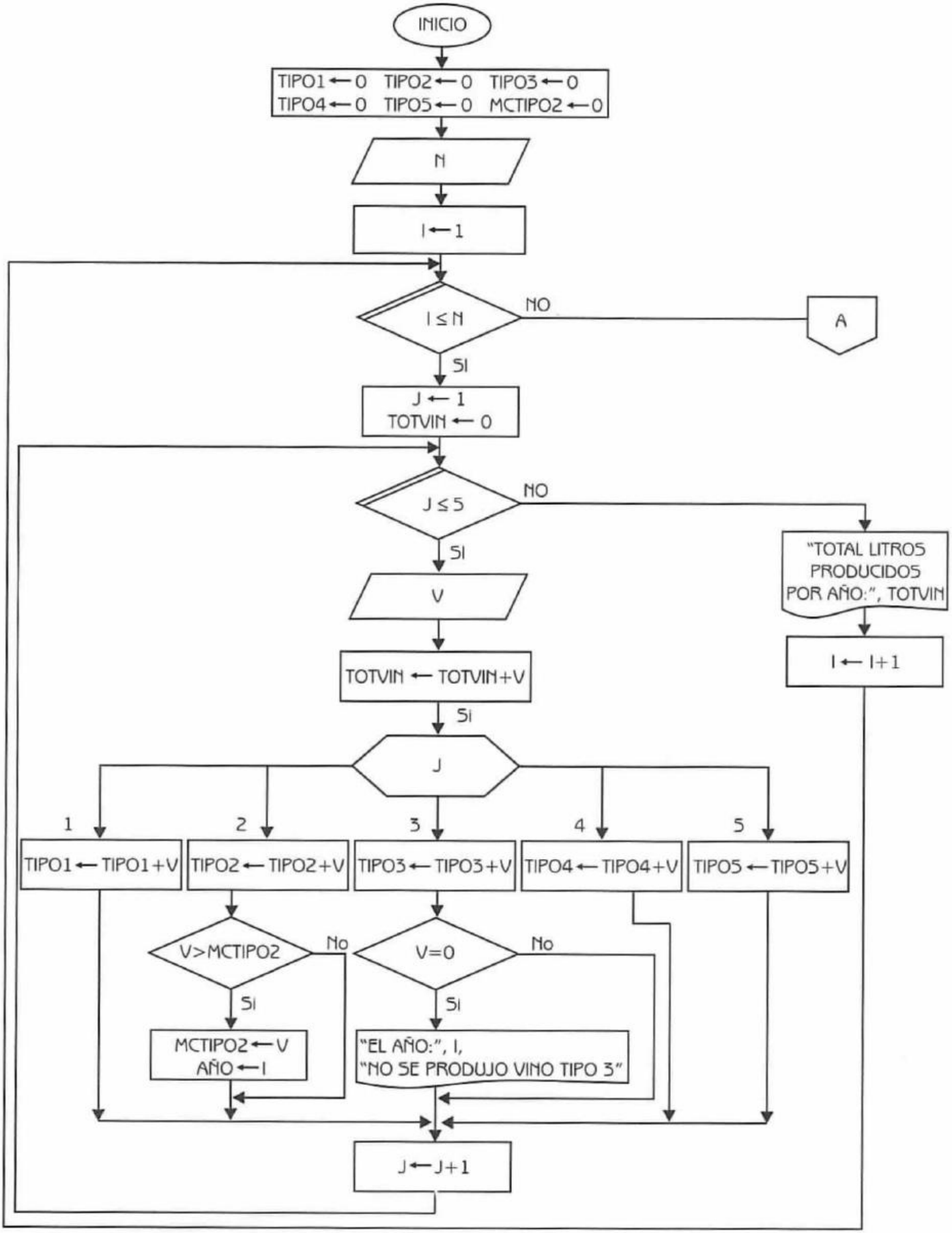


Diagrama de Flujo 3.23 (*continúa*)

www.FreeLibros.com

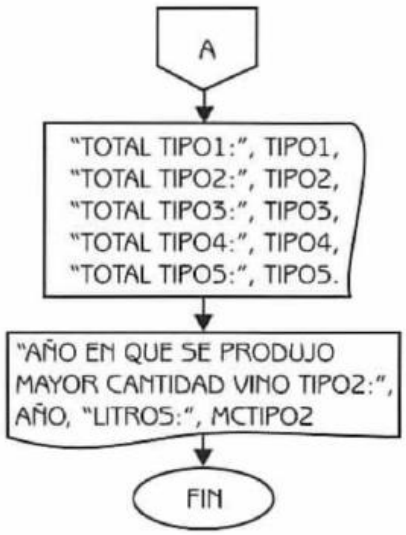


Diagrama de Flujo 3.23 (*continuación*)

Explicación de las Variables

I: Variable de tipo entero. Representa la variable de control del ciclo externo.

TIPO1, TIPO2, TIPO3, TIPO4 y TIPO5: Variables de tipo real. Acumulan el total de litros de vino producidos en los N años de los tipos 1,2,3,4 y 5, respectivamente.

MCTIPO2: Variable de tipo real. Acumula la cantidad de vino tipo 2 producido en el mayor año de producción. Localiza un máximo, por eso se inicializa con un valor mínimo (cero en este caso).

AÑO: Variable de tipo entero. Almacena el año en que se presenta la mayor producción, del vino tipo2.

J: Variable de tipo entero. Representa al contador del ciclo interno.

N: Variable de tipo entero.

TOTVIN: Variable de tipo real. Acumula el total de vinos producido en el año.

V: Variable de tipo real.

A continuación presentamos el programa correspondiente.

Programa 3.22

```
BODEGA
{El programa dada información sobre los vinos producidos por una bodega, obtiene información estadística de utilidad}
{I, AÑO, J y N son variables de tipo entero. TIPO1, TIPO2, TIPO3, TIPO4, TIPO5, MCTIPO2, TOTVIN y V son variables de tipo real}
```

www.FreeLibros.com

```
1. Hacer TIPO1 ⟵ 0, TIPO2 ⟵ 0, TIPO3 ⟵ 0, TIPO4 ⟵ 0, TIPO5 ⟵ 0
        y MCTIPO2 ⟵ 0
2. Leer N
3. Hacer I ⟵ 1
4. Repetir con I desde 1 hasta N
        Hacer TOTVIN ⟵ 0 y J ⟵ 1
    4.1 Repetir con J desde 1 hasta 5
            Leer V
            Hacer TOTVIN ⟵ TOTVIN + V
        4.1.1 Si J igual
            1 : Hacer TIPO1 ⟵ TIPO1 + V
            2 : Hacer TIPO2 ⟵ TIPO2 + V
                4.1.1.1 Si V > MCTIPO2 entonces
                        Hacer MCTIPO2 ⟵ V y AÑO ⟵ I
                4.1.1.2 {Fin del condicional del paso 4.1.1.1}
            3 : Hacer TIPO3 ⟵ TIPO3 + V
                4.1.1.3 Si (V=0) entonces
                        Escribir " En el año : " , I, " No se produjo vino TIPO3 "
                4.1.1.4 {Fin del condicional del paso 4.1.1.3}
            4 : Hacer TIPO4 ⟵ TIPO4 + V
            5 : Hacer TIPO5 ⟵ TIPO5 + V
        4.1.2 {Fin del condicional del paso 4.1.1}
            Hacer J ⟵ J + 1
    4.2 {Fin del ciclo del paso 4.1}
        Escribir " Total de litros producidos por año: ", TOTVIN
        Hacer I ⟵ I + 1
5. {Fin del ciclo del paso 4}
6. Escribir   " TOTAL TIPO1 : ", TIPO 1,
              " TOTAL TIPO2 : ", TIPO2,
              " TOTAL TIPO3 : ", TIPO3,
              " TOTAL TIPO4 : ", TIPO4,
              " TOTAL TIPO5 : ", TIPO5,
7. Escribir " AÑO en que se produjo mayor cantidad vino TIPO2 : ", AÑO,
              " LITROS : ", MCTIPO2
```

Problema 3.17

Se dice que un entero positivo N es un número primo si los únicos enteros positivos que lo dividen son exactamente 1 y N. Diseñe un diagrama de flujo que lea un número M, y obtenga y cuente todos los números primos menores a M.

Dato: M (variable de tipo entero que representa el número límite que se ingresa).

www.FreeLibros.com

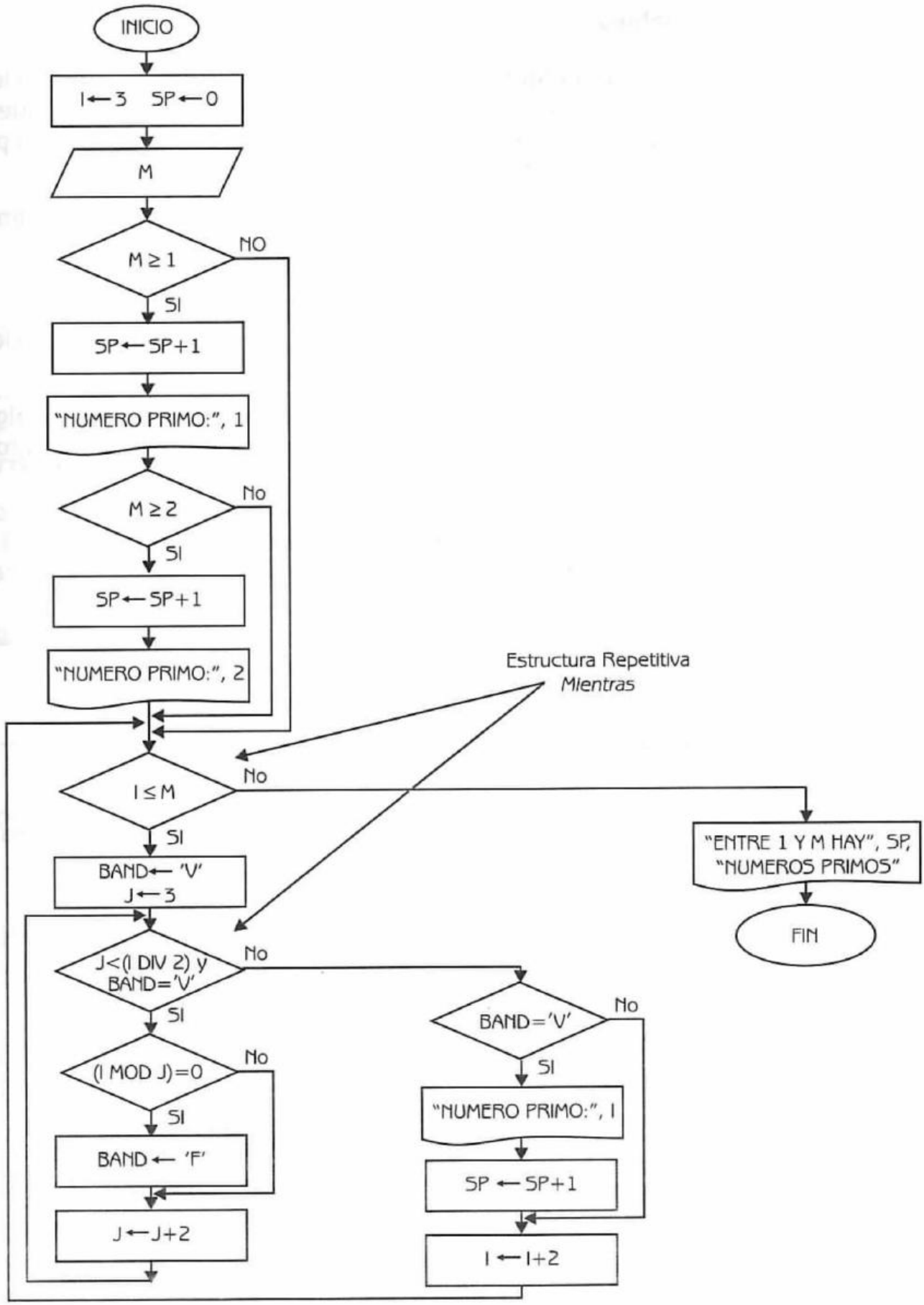


Diagrama de Flujo 3.24

www.FreeLibros.com

Explicación de las variables

I: Variable de tipo entero. Representa la variable de control del ciclo externo. Se inicializa en 3 porque sabemos por definición que el número 1 y los números pares (con excepción del 2) no son primos, y se incrementa de 2 en 2.

SP: Variable de tipo entero. Acumula el número de números primos comprendidos entre 1 y M.

M: Variable de tipo real.

J: Variable de tipo entero. Representa la variable de control del ciclo interno. Se inicializa en 3 y se incrementa de 2 en 2.

BAND: Variable de tipo caracter. Se inicializa en verdadero ('V'). Si alguna de las divisiones que se realizan produce módulo cero, entonces toma el valor de falso ('F').

A continuación, presentamos el programa correspondiente.

Programa 3.23

```
PRIMOS

{El programa, dado un entero positivo M, obtiene e imprime la cantidad de números primos comprendidos entre 1 y M}

{I, SP y J son variables de tipo entero. M es una variable de tipo real. BAND es una variable de tipo caracter}

1. Hacer I ←—— 3 y SP ←—— 0
2. Leer M
3. Si M ≥ 1 entonces
       Hacer SP ←—— SP + 1
       Escribrir "Número Primo:", 1
   3.1 Si M > 2 entonces
           Hacer SP ←—— SP + 1
           Escribe "Número Primo:", 2
   3.2 {Fin del condicional del paso 3.1}
4. {Fin del condicional del paso 3}
5. Mientras I ≤ M Repetir
           Hacer BAND ←—— 'V' y J ←—— 3
   5.1 Mientras (J < (I DIV 2)) y (BAND='V') Repetir
```

www.FreeLibros.com

```
        5.1.1 Si (I MOD J)=0 entonces
                  Hacer BAND ⟵ 'F'
        5.1.2 {Fin del condicional del paso 5.1.1}
                  Hacer J ⟵ J+2
    5.2 {Fin del ciclo del paso 5.1}
    5.3 Si (BAND='V') entonces
            Escribir "Número primo:", I
            Hacer SP⟵ SP+1
    5.4 {Fin del condicional del paso 5.3}
        Hacer I ⟵ I+2
6. {Fin del ciclo del paso 5}
4. Escribe "ENTRE 1 y M hay", SP, "NUMEROS PRIMOS"
```

Problema 3.18

Los datos reunidos en la Secretaría de Industrias relacionado a la producción de N fábricas ($N \leq 1000$) en cada uno de los meses del año anterior, se proporcionan de la siguiente forma:

Datos: N
$FABRICA_1, MES_{1,1}, MES_{1,2}, \ldots, MES_{1,12}$
$FABRICA_2, MES_{2,1}, MES_{2,2}, \ldots, MES_{2,12}$
. . .
$FABRICA_N, MES_{N,1}, MES_{N,2}, \ldots, MES_{N,12}$

Donde:

N	es una variable de tipo entero que indica el número de fábricas registradas en la Secretaría de Industrias ($N \leq 100$).
$FABRICA_i$	es una variable de tipo entero que indica la clave que identifica a la fábrica$_i$ ($1 \leq i \leq N$).
$MES_{i,j}$	es una variable de tipo real que representa la producción de la fábrica i en el mes j ($1 \leq i \leq N, 1 \leq j \leq 12$).

Haga un diagrama de flujo que calcule lo siguiente:

a) Los totales anuales de producción de cada fábrica.

b) La clave de la fábrica que más produjo en el año. Indicar también el total de la producción.

c) Imprimir las claves de las fábricas cuyas producciones en el mes de julio superaron los $ 3 000 000.

www.FreeLibros.com

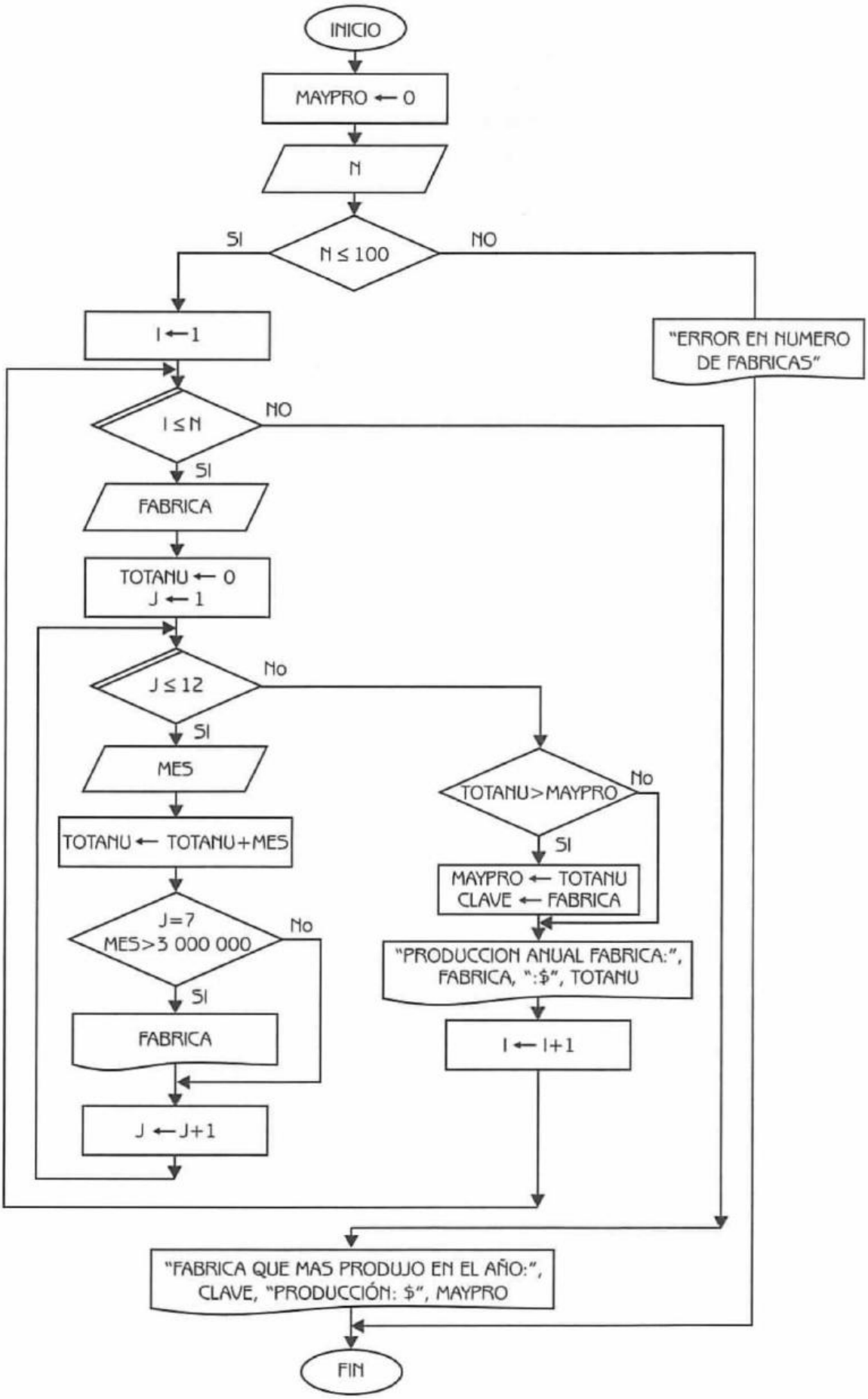


Diagrama de Flujo 3.25

www.FreeLibros.com

Explicación de las variables

I: Variable de tipo entero. Representa la variable de control del ciclo externo.

MAYPRO: Variable de tipo real. Almacena el total de la fábrica que más produjo en el año. Se utiliza para localizar un máximo, por eso se inicializa en cero.

CLAVE: Variable de tipo entero. Almacena la clave de la fábrica que más dinero produjo en el año.

J: Variable de tipo entero. Representa la variable de control del ciclo interno.

TOTANU: Variable de tipo real. Acumula lo que vendió la fábrica en el año.

N y *FABRICA*: Variables de tipo entero.

MES: Variable de tipo real.

Programa 3.24

```
SECRETARIA_DE_INDUSTRIAS

{El programa, dada cierta información que recibe de la Secretaría de Industrias relacionado a la producción de N fábricas, obtiene información estadística de la misma}

{I, CLAVE, J, N y FABRICA son variables de tipo entero. MAYPRO, TOTANU y MES son variables de tipo real}

1. Hacer MAYPRO ←— 0
2. Leer N
3. Si N ≤ 1000
        entonces
        Hacer I ←— 1
        3.1 Repetir con I desde 1 hasta N
                Leer FABRICA
                Hacer TOTANU ←— 0 y J ←— 1
            3.1.1 Repetir con J desde 1 hasta 12
                    Leer MES
                    Hacer TOTANU ←— TOTANU + MES
```

www.FreeLibros.com

```
                3.1.1.1 Si (J = 7) y (MES > 3 000 000) entonces
                            Escribir FABRICA
                3.1.1.2 {Fin del condicional del paso 3.1.1.1}
                            Hacer J ←—— J + 1
        3.1.2 {Fin del ciclo del paso 3.1.1}
        3.1.3 Si TOTANU > MAYPRO entonces
                    Hacer MAYPRO ←—— TOTANU y CLAVE ←—— FABRICA
        3.1.4 {Fin del condicional del paso 3.1.3}
              Escribir  " Producción anual fábrica : ", FABRICA,
                        " : $ ", TOTANU
              Hacer I ←—— I + 1
    3.2 {Fin del ciclo del paso 3.1}
        Escribir  " FABRICA que más produjo en el año : ",CLAVE,
                  " PRODUCCION : $ ", MAYPRO
    sino
        Escribir  " ERROR EN NUMERO DE FABRICAS "
4. {Fin del condicional del paso 3}
```

Problema 3.19

Un número es perfecto si "la suma de sus divisores excepto el mismo es igual al propio número". Haga un diagrama de flujo para calcular e imprimir los números perfectos menores o iguales que N.

Dato: N (variable de tipo entero que representa el límite de los números naturales enteros que se probarán para determinar si son perfectos).

www.FreeLibros.com

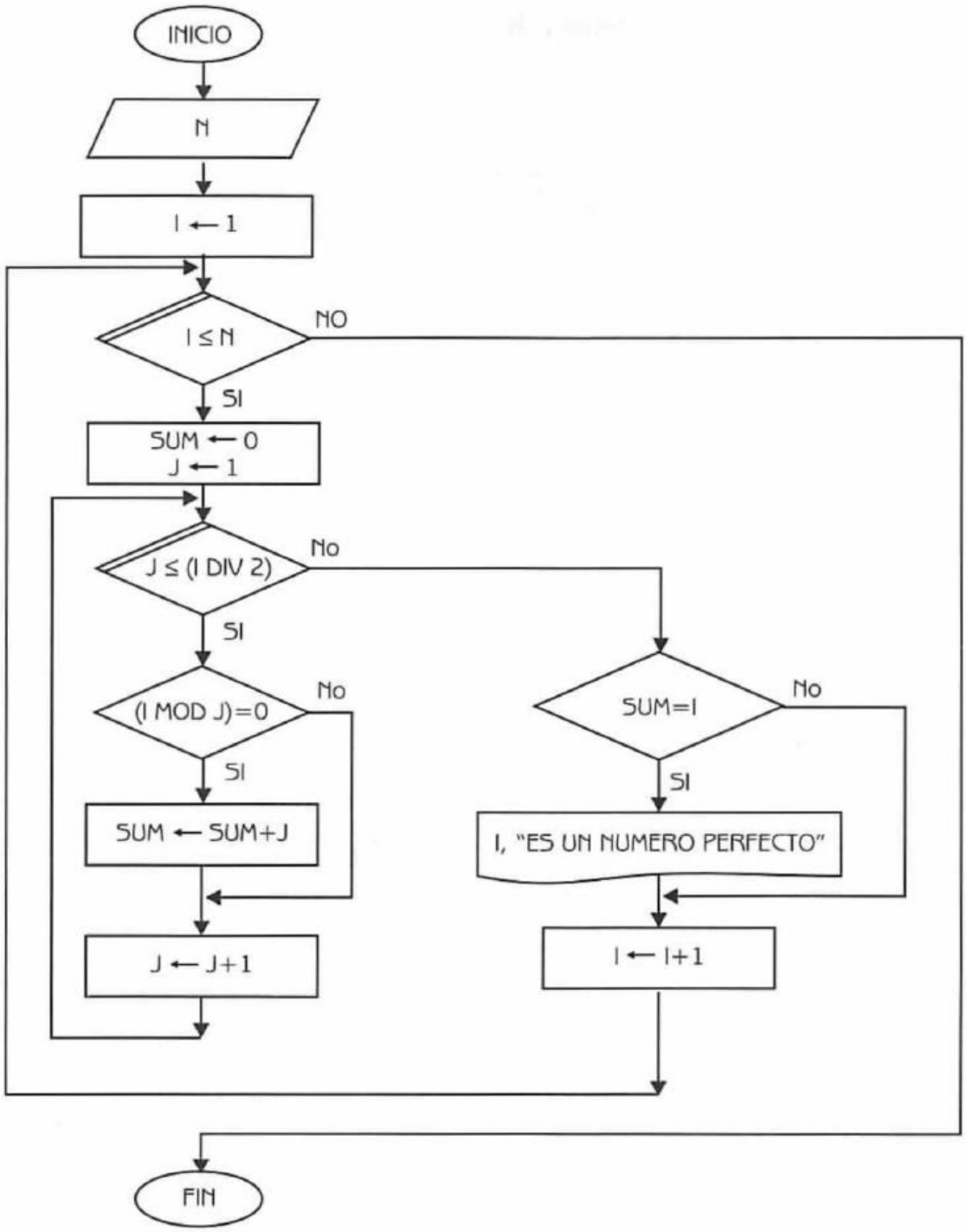


Diagrama de Flujo 3.26

Explicación de las variables

I: Es una variable de tipo entero. Representa la variable de control del ciclo externo.
N: Variable de tipo entero.
SUM: Variable de tipo entero. Acumula los divisores para posteriormente probar si el número es perfecto.
J Variable de tipo entero. Representa la variable de control del ciclo interno.

www.FreeLibros.com

A continuación en la siguiente tabla, podemos observar el seguimiento del diagrama de flujo:

Tabla 3.18

N	I	SUM	J	(IMPRESION)
9	1			
	2	0	1	
		0	1	
	3	1	2	
		0	1	
	4	1	2	
		0	1	
		1	2	
	5	3	3	
		0	1	
		1	2	
	6		3	
		0	1	
		1	2	
		3	3	
	7	6	4	6 ES UN NUMERO PERFECTO
		0	1	
		1	2	
			3	
	8		4	
		0	1	
		1	2	
		3	3	
		7	4	
	9		5	
		0	1	
		1	2	
		4	3	
			4	
	10		5	

: Expresa valores que se imprimen.

www.FreeLibros.com

Programa 3.25

```
NUMEROS_PERFECTOS

{El programa dado un número entero N, obtiene e imprime los números perfectos
comprendidos entre 1 y N}

{I,N,SUM y J son variables de tipo entero}

1. Leer N
2. Hacer I ←—— 1
3. Repetir con I desde 1 hasta N
        Hacer SUM ←—— 0 y J ←—— 1
   3.1 Repetir con J desde 1 hasta (I DIV 2)
       3.1.1 Si (I MOD J) = 0 entonces
                    Hacer SUM ←—— SUM + J
       3.1.2 {Fin del condicional del paso 3.1.1}
             Hacer J ←—— J+1
   3.2 {Fin del ciclo del paso 3.1}
   3.3 Si SUM = I entonces
           Escribir I, " Es un número perfecto "
   3.4 {Fin del condicional del paso 3.3}
       Hacer I←—— I + 1
4. {Fin del ciclo del paso 3}
```

www.FreeLibros.com

4

Estructuras de datos: arreglos

4.1 Introducción

Con frecuencia podemos encontrar problemas cuya solución es muy difícil de implementar si utilizamos *tipos simples de datos*. Por otra parte, podemos encontrar una buena solución al problema utilizando *tipos estructurados de datos*.

Con el objeto de ilustrar este concepto, presentaremos un problema y dos soluciones al mismo utilizando tipos simples de datos. El lector observará lo complejo que resulta un algoritmo de solución para ciertos problemas si no utilizamos tipos estructurados de datos. Posteriormente y luego de presentar los arreglos, mostraremos una solución al problema planteado utilizando tipos estructurados de datos.

Ejemplo 4.1

Se tienen los sueldos de un grupo de 70 empleados de una empresa y necesitamos saber cuántos de estos empleados tienen un sueldo superior al promedio del grupo.

Cómo podríamos resolver este problema? A continuación presentamos la primera solución.

Datos: $S_1, S_2, \ldots, S_{70}$
Donde: Si representa el sueldo del empleado i.

www.FreeLibros.com

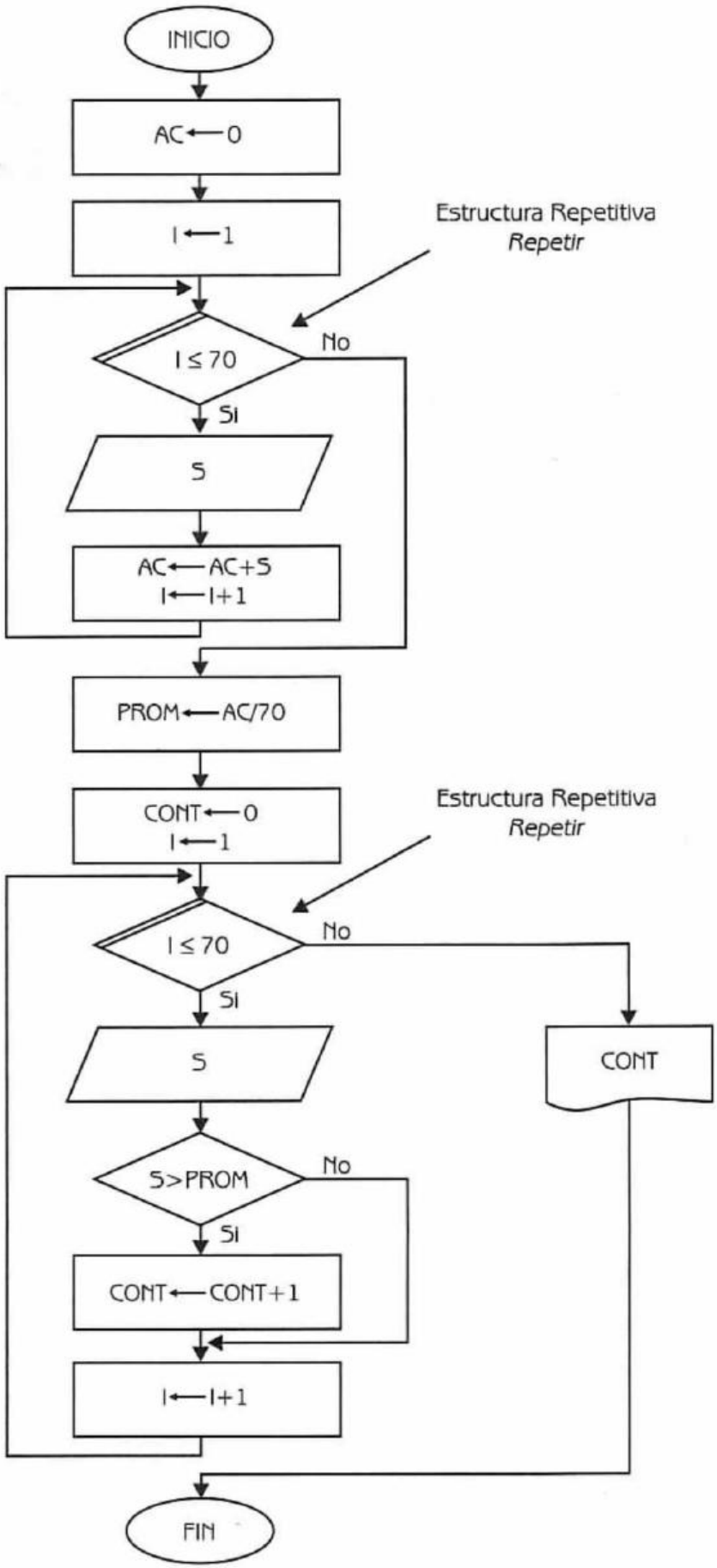


Diagrama de Flujo 4.1

www.FreeLibros.com

Nota: Como necesitamos saber cuántos empleados ganaron un sueldo superior al promedio, se tuvieron que volver a leer los 70 sueldos para comparar cada uno de ellos con el promedio.

Explicación de las Variables

I: Variable de tipo entero. Representa la variable de control del ciclo.

S: Variable de tipo real.

AC: Variable de tipo real. Acumula los sueldos de los empleados.

PROM: Variable de tipo real. Almacena el promedio de los sueldos.

CONT: Variable de tipo entero. Se utiliza para contar cuántos sueldos son superiores al promedio.

En la tabla 4.1. podemos observar el seguimiento del diagrama de flujo. Por razones de practicidad en lugar de considerar 70 sueldos, consideraremos solamente 5.

Tabla 4.1

I	AC	S	PROM	CONT
1	0			
2	1200.00	1200.00		
3	3950.00	2750.00		
4	7575.00	3625.00		
5	8555.00	980.00		
6	9605.70	1050.70		
1			1921.14	0
2		1200.00		
3		2750.00		1
4		3625.00		2
5		980.00		
6		1050.70		

: Expresa el valor que se imprime.

El Diagrama de flujo 4.1 en lenguaje algorítmico, se expresa de esta forma:

www.FreeLibros.com

Programa 4.1

```
DOBLE LECTURA

{El programa resuelve el problema planteado en el ejemplo 4.1 por medio de una
doble lectura. I y CONT son variables de tipo entero. S, AC y PROM son variables de
tipo real}.

1. Hacer AC ←— 0 e I ←— 1
2. Repetir con I desde 1 hasta 70
        Leer S
        Hacer AC ←— AC + S e I ←— I + 1
3. {Fin del ciclo del paso 2}
4. Hacer PROM ←— AC/70, CONT ←— 0 e I ←—1
5. Repetir con I desde 1 hasta 70
        Leer S
    5.1 Si S > PROM entonces
            Hacer CONT ←— CONT + 1
    5.2 {Fin del condicional del paso 5.1.}
        Hacer I ←— I + 1
6. {Fin del ciclo del paso 5}
7. Escribir CONT.
```

Ejemplo 4.2

A continuación presentamos la segunda solución.

Datos: S1, S2, S3, S4,..., S70

Donde:

S1, S2, S3, S4, ... , S70	son variables de tipo real que representan los sueldos de los 70 empleados de la empresa.

www.FreeLibros.com

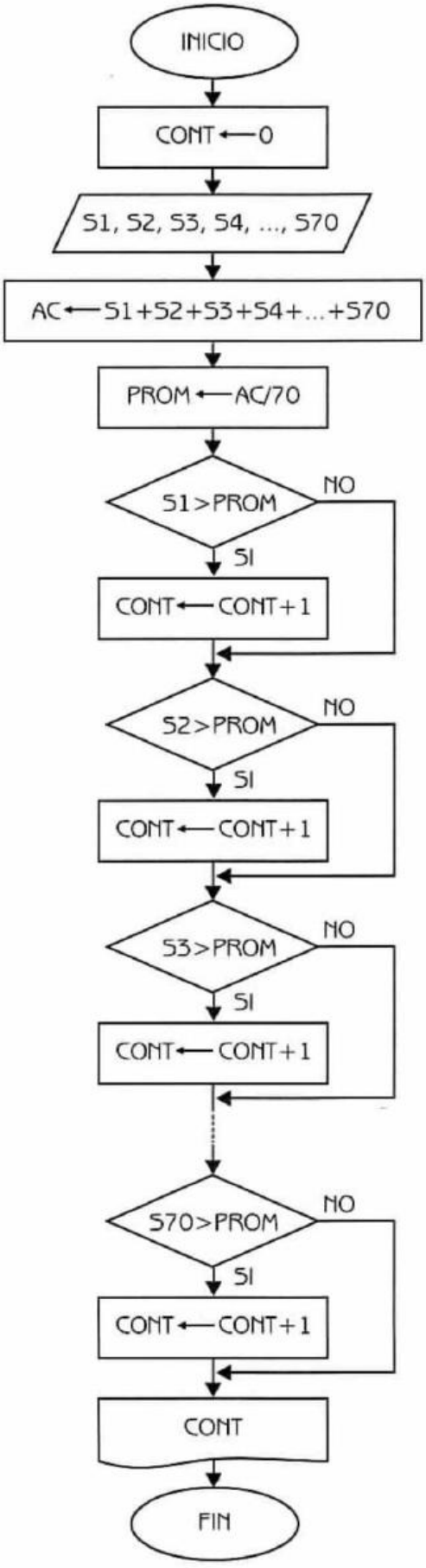


Diagrama de Flujo 4.2

www.FreeLibros.com

Explicación de las variables:

CONT:	Variable de tipo entero. Se utiliza para contar los sueldos que son superiores al promedio.
AC:	Variable de tipo real. Se utiliza para almacenar la suma de los 70 sueldos.
PROM:	Variable de tipo real. Se utiliza para almacenar el promedio de los 70 sueldos.
S1, S2, S3,..., S70:	Variables de tipo real.

Programa 4.2

```
MUCHAS VARIABLES

{El programa resuelve el problema planteado en el ejemplo 4.1, utilizando muchas
variables}

{CONT es una variable de tipo entero. AC, PROM, S1, S2, S3, ..., S70 son variables
de tipo real}

1. Hacer CONT ←— 0
2. Leer S1, S2, S3, S4,..., S70
3. Hacer AC ←— S1 + S2 + S3 + S4 ...+ S70  y  PROM ←— AC/70
4. Si S1 > PROM entonces
       Hacer CONT ←— CONT + 1
5. {Fin del condicional del paso 4}
6. Si S2 > PROM entonces
       Hacer CONT ←— CONT + 1
7. {Fin del condicional del paso 6}
8. Si S3 > PROM entonces
       Hacer CONT ←— CONT + 1
9. {Fin del condicional del paso 8}
...
142. Si S70 > PROM entonces
       Hacer CONT ←— CONT + 1
143. {Fin del condicional del paso 142}
144. Escribir CONT
```

Las dos soluciones son muy representativas de los inconvenientes a los que debemos enfrentarnos al tratar de resolver el problema utilizando tipos simples de datos. En la solución planteada en el diagrama de flujo 4.1 el usuario del algo-

www.FreeLibros.com

ritmo debe ingresar dos veces el conjunto de datos. Esto resulta totalmente impráctico (considere que el número de datos puede ser mayor que 70) y además ineficiente (la operación de lectura ya sea de manera interactiva con el usuario o desde un archivo debe repetirse, lo que ocasiona pérdida de tiempo).

En la solución planteada en el diagrama de flujo 4.2. se manejan 70 variables en memoria. Esta solución presenta el inconveniente de que el manejo de las variables puede volverse incontrolable, si el número de las mismas crece considerablemente. Además algunos pasos especificados en el algoritmo se repiten ya que no pueden generalizarse. Esta característica no sólo ocasiona más trabajo, sino también es causa de errores. Es sabido que ejecutar una tarea repetidamente (en este caso escribir un mismo paso varias veces) quita interés a la acción que se está llevando a cabo, lo que a su vez ocasiona errores. Puede observarse entonces, que ninguna de las dos soluciones resulta práctica y eficiente. Es necesario un nuevo tipo de datos que permita tratar estos problemas de una manera más adecuada. Los tipos de datos estructurados que ayudan a resolver problemas como éste son los arreglos.

4.2 Arreglos unidimensionales

Un arreglo se define como una colección finita, homogénea y ordenada de elementos.

Finita: todo arreglo tiene un límite, es decir se debe determinar cuál será el número máximo de elementos que podrán formar parte del arreglo.

Homogénea: todos los elementos de un arreglo son del mismo tipo (todos enteros, todos reales, etc., pero nunca una combinación de distintos tipos).

Ordenada: se puede determinar cuál es el primer elemento, el segundo, el tercero,... y el n–ésimo elemento.

Un arreglo puede representarse gráficamente como se muestra en la figura 4.1.

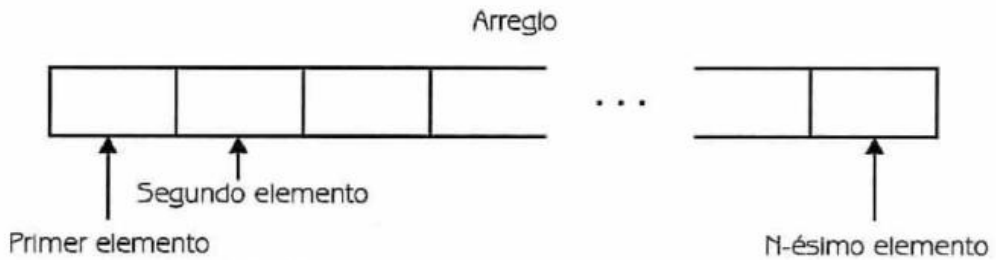


Figura 4.1 Representación de arreglos.

Un arreglo tiene la característica de que puede almacenar a N elementos del mismo tipo y además permite el acceso a cada uno de estos elementos. Así, se distinguen dos partes en los arreglos:

www.FreeLibros.com

- los componentes.
- los índices.

Los componentes hacen referencia a los elementos que *componen* o forman el arreglo. Es decir, son los valores que se almacenan en cada una de sus casillas. Los índices, por otra parte, son los que permiten accesar a los componentes del arreglo en forma individual. Para hacer referencia a un componente de un arreglo se necesita:

- el nombre del arreglo.
- el índice del elemento.

En la figura 4.2 representamos un arreglo y se indican sus componentes y sus índices.

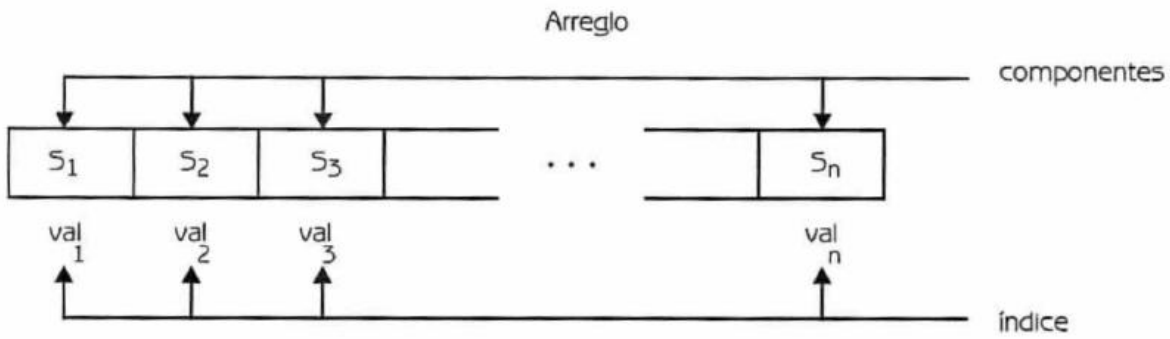


Figura 4.2 Indices y componentes de un arreglo.

4.2.1 Definición de arreglos

Como no es nuestra intención seguir la sintaxis de algún lenguaje de programación en particular, definimos un arreglo de la siguiente manera:

ident_arreglo = ARREGLO [líminf .. límsup] DE tipo

Con los valores líminf y límsup se declara el tipo de los índices así como el número de elementos que tendrá el arreglo. El número total de elementos (NTE) que tendrá el arreglo puede calcularse con la fórmula 4.1:

$$NTE = límsup - líminf + 1$$

Fórmula 4.1

www.FreeLibros.com

Con **tipo** se declara el tipo de datos para todos los elementos del arreglo. El tipo de los elementos no tiene que ser necesariamente el mismo que el de los índices.

Observaciones:

a) El tipo del índice puede ser cualquier tipo ordinal (caracter, entero, etc.).
b) El tipo de los componentes puede ser cualquier tipo (entero, real, cadena de caracteres, registro, arreglo, etc.).
c) Se utilizan los corchetes "[]" para indicar el índice de un arreglo. Entre los [] se debe escribir un valor ordinal (puede ser una variable, una constante o una expresión tan compleja como se quiera, pero que dé como resultado un valor ordinal).

Se verán a continuación algunos ejemplos de arreglos:

Ejemplo 4.3

Sea ARRE un arreglo de 70 elementos enteros con índices enteros. Su representación queda como se muestra en la figura 4.3.

ARRE = ARREGLO [1. .70] DE enteros

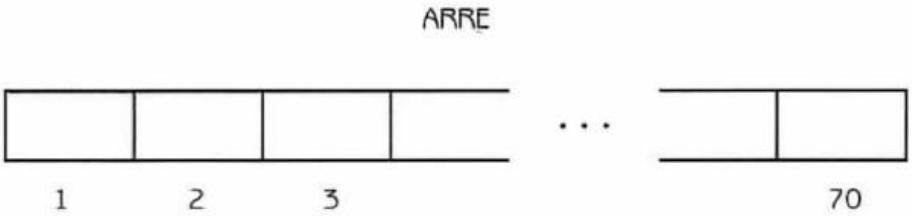


Figura 4.3

- NTE = (70 – 1 + 1) = 70
- Cada elemento del arreglo ARRE será un número entero y podrá accesarse por medio de un índice que será un valor comprendido entre 1 y 70.

Así por ejemplo:

ARRE[1] hace referencia al elemento de la posición 1.
ARRE[2] hace referencia al elemento de la posición 2.
. . .
. . .
ARRE[70]hace referencia al elemento de la posición 70.

www.FreeLibros.com

Ejemplo 4.4

Consideremos el arreglo de enteros del ejemplo anterior. Suponiendo que el límite inferior es igual a –13 y el límite superior igual a 56, su representación queda como se muestra en la figura 4.4.

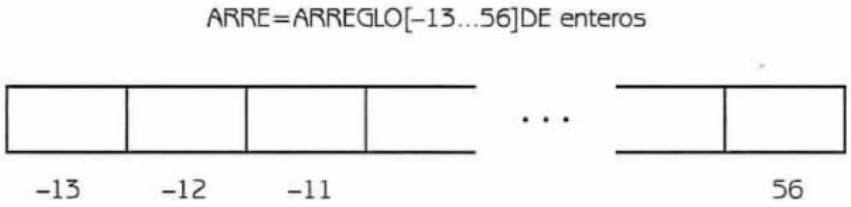


Figura 4.4

- NTE = (56–(–13)+1) = 70
- Cada elemento del arreglo será un número entero y podrá accesarse por medio de un índice que será un valor comprendido entre –13 y 56.

Así por ejemplo:

ARRE[–13] hace referencia al primer elemento del arreglo.
ARRE[–12] hace referencia al segundo elemento del arreglo.
. . .
ARRE[1] hace referencia al decimoquinto elemento del arreglo.
. . .
ARRE[56] hace referencia al último elemento del arreglo.

4.2.2 Operaciones con arreglos

A continuación se presentan las operaciones más comunes en arreglos:

- Lectura/Escritura.
- Asignación.
- Actualización: Inserción.
  Eliminación.
  Modificación.
- Ordenación.
- Búsqueda.

Como los arreglos son datos estructurados, muchas de estas operaciones no pueden llevarse a cabo de manera global, sino que se debe trabajar sobre cada elemento.

A continuación se analizará cada una de estas operaciones. Ordenación y búsqueda, serán analizadas en los problemas que presentaremos posteriormente.

www.FreeLibros.com

Lectura

El proceso de lectura de un arreglo consiste en leer y asignar un valor a cada uno de sus elementos. Supóngase que se desea leer todos los elementos del arreglo ARRE presentado en el ejemplo 4.3, en forma consecutiva. Podría hacerse de la siguiente manera:

Leer ARRE[1], ARRE[2], . . . ,ARRE[70]

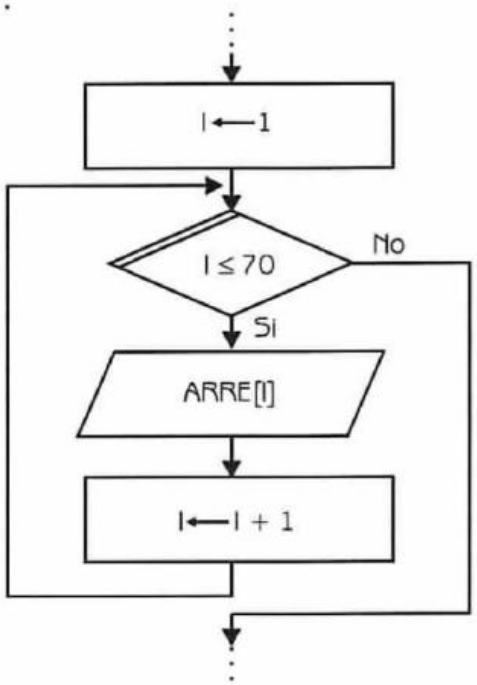
Leer ARRE[2],

. . .

Leer ARRE[70]

De esta forma no resulta práctico, por lo tanto se usará un ciclo para leer todos los elementos del arreglo.

```
.
.
.
Hacer I ⟵ 1
Repetir con I desde 1 hasta 70
        Leer ARRE[I]
        Hacer I ⟵ I + 1
{Fin del ciclo}
.
.
.
```



Al variar el valor de I, cada elemento leído se asigna al correspondiente componente del arreglo según la posición indicada por I.

www.FreeLibros.com

Para I = 1, se lee ARRE[1]
I = 2, se lee ARRE[2]
. . .
I = 70, se lee ARRE[70]

Al finalizar el ciclo de lectura se tendrá asignado un valor a cada uno de los componentes del arreglo ARRE. El arreglo queda como se muestra en la figura 4.5.

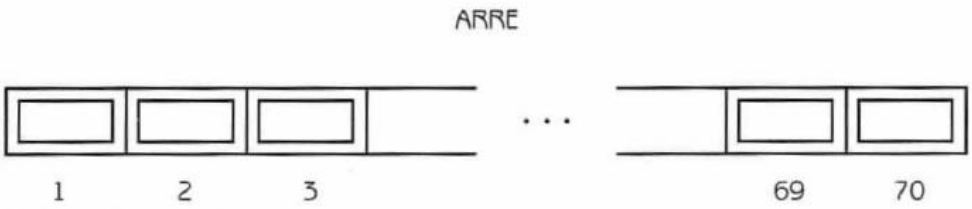
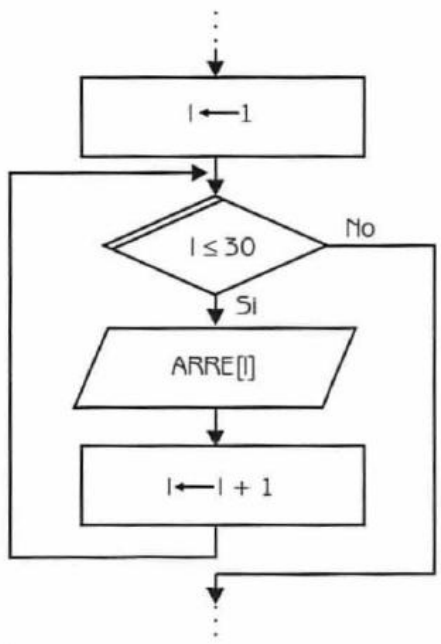


Figura 4.5 Lectura de arreglos.

Puede suceder que no se necesite leer todos los componentes, sino solamente algunos de ellos. Supóngase por ejemplo que deben leerse los elementos con índices comprendidos entre el 1 y el 30 inclusive. El ciclo necesario es el siguiente:




```
Hacer I ←—— 1
Repetir con I desde 1 hasta 30
        Leer ARRE[I]
        Hacer I ←—— I + 1
{Fin del ciclo}
```

www.FreeLibros.com

El arreglo queda como se muestra en la figura 4.6.

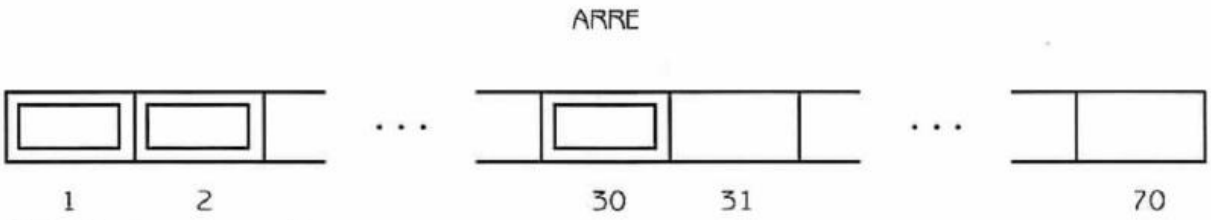
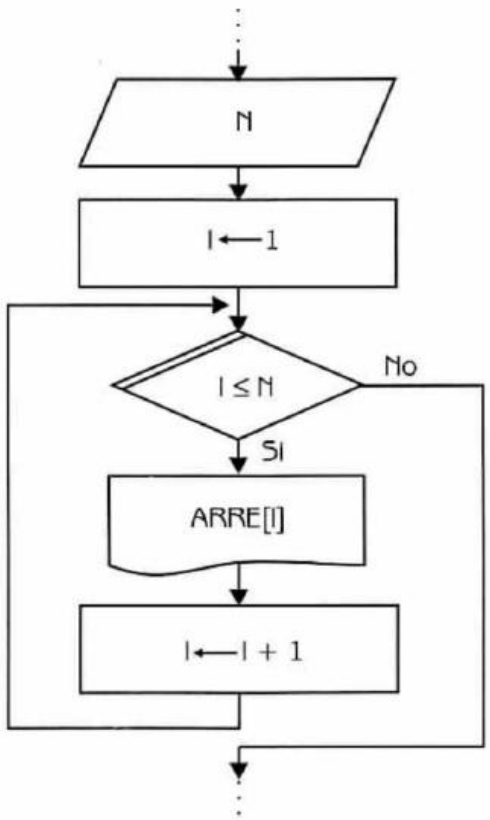


Figura 4.6 Lectura de arreglos.

Escritura

El caso de escritura es similar al de lectura. Se debe escribir el valor de cada uno de los componentes. Supóngase que se desea escribir los primeros N componentes del arreglo ARRE (ejemplo 4.3) en forma consecutiva. Los pasos a seguir son los siguientes:




```
.
.
Leer N
Hacer I ←—— 1
Repetir con I desde 1 hasta N
        Escribir ARRE[I]
        Hacer I ←—— I + 1
{Fin del ciclo}
.
.
.
```

www.FreeLibros.com

Al variar el valor de I se escribe el elemento de ARRE correspondiente a la posición indicada por I.

Para I = 1, se escribe el valor de ARRE[1]
I = 2, se escribe el valor de ARRE[2]
. . .
I = N, se escribe el valor de ARRE[N]

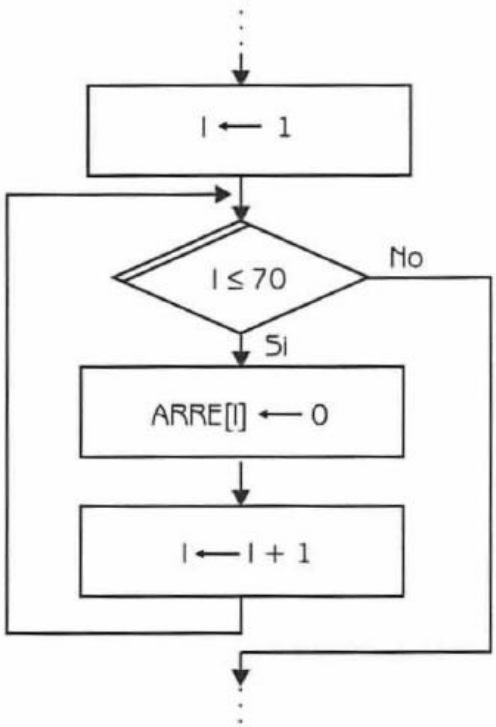
Asignación

En general no es posible asignar directamente un valor a todo el arreglo, sino que se debe asignar el valor deseado a cada componente. En seguida se analizan algunos ejemplos de asignación.

En los dos primeros casos se asigna un valor a una determinada casilla del arreglo (en el primero a la señalada por el índice 1 y en el segundo a la indicada por el índice 3).

ARRE [1] ⟵ 120
ARRE [3] ⟵ ARRE [1]/4

En el tercer caso se asigna el 0 a todas las casillas del arreglo, con lo que éste queda como se muestra en la figura 4.7.



www.FreeLibros.com

```
.
.
.
Hacer I ⟵ 1
Repetir con I desde 1 hasta 70
        Hacer ARRE [I] ⟵ 0 e I ⟵ I + 1
{Fin del ciclo}
.
.
.
```

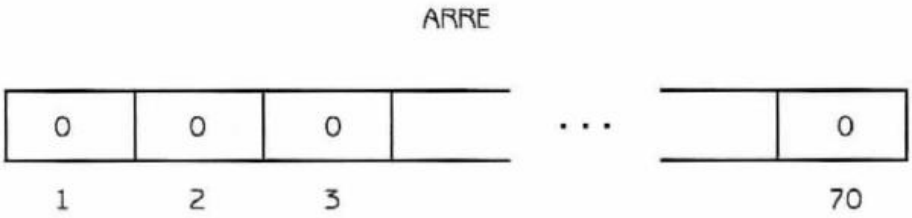
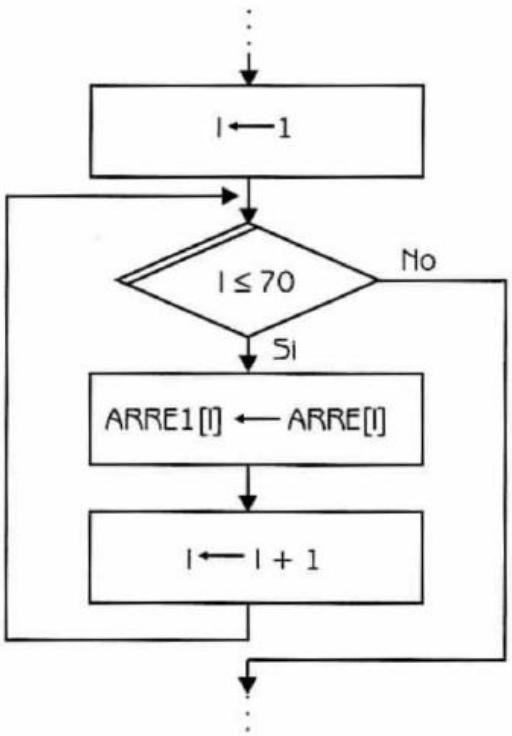


Figura 4.7 Asignación de arreglos.

En algunos lenguajes, por otra parte, es posible asignar una variable tipo arreglo a otra exactamente del mismo tipo.

```
ARRE1 ⟵ ARRE
```

La expresión anterior es equivalente a:



www.FreeLibros.com

```
.
.
Hacer I ←—— 1
Repetir con I desde 1 hasta 70
        Hacer ARRE1[I] ←—— ARRE[I] e I ←—— I + 1
{Fin del ciclo}
.
.
```

Actualización

En un arreglo se pueden insertar, eliminar y/o modificar elementos. Para llevar a cabo estas operaciones eficientemente se debe tener en cuenta si el arreglo está ordenado o desordenado. Es decir, si sus componentes respetan algún orden entre sí. Las operaciones de inserción, eliminación y modificación serán tratadas separadamente para arreglos desordenados y ordenados.

a) ARREGLOS DESORDENADOS

Considere un arreglo A de 100 elementos como el presentado en la figura 4.8.

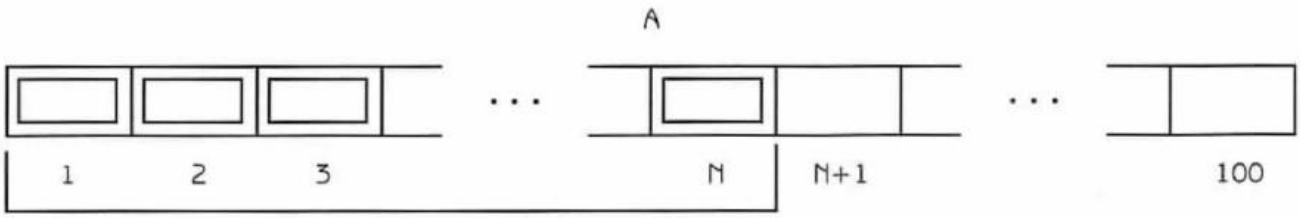


Figura 4.8 Actualización de arreglos desordenados.

La figura indica que los primeros N elementos tienen asignado un valor.

a.1) Inserción: Para insertar un elemento Y en un arreglo A desordenado debe verificarse que exista espacio. Si se cumple esta condición, entonces se asignará a la posición N + 1 el nuevo elemento. A continuación presentamos el diagrama de flujo (4.3) correspondiente.

Explicación de las variables

N: Variable de tipo entero. Almacena el número actual de elementos del arreglo.
Y: Variable de tipo entero. Representa al valor que se va a insertar.
A: Arreglo unidimensional de tipo entero. Su capacidad máxima es de 100 elementos.

www.FreeLibros.com

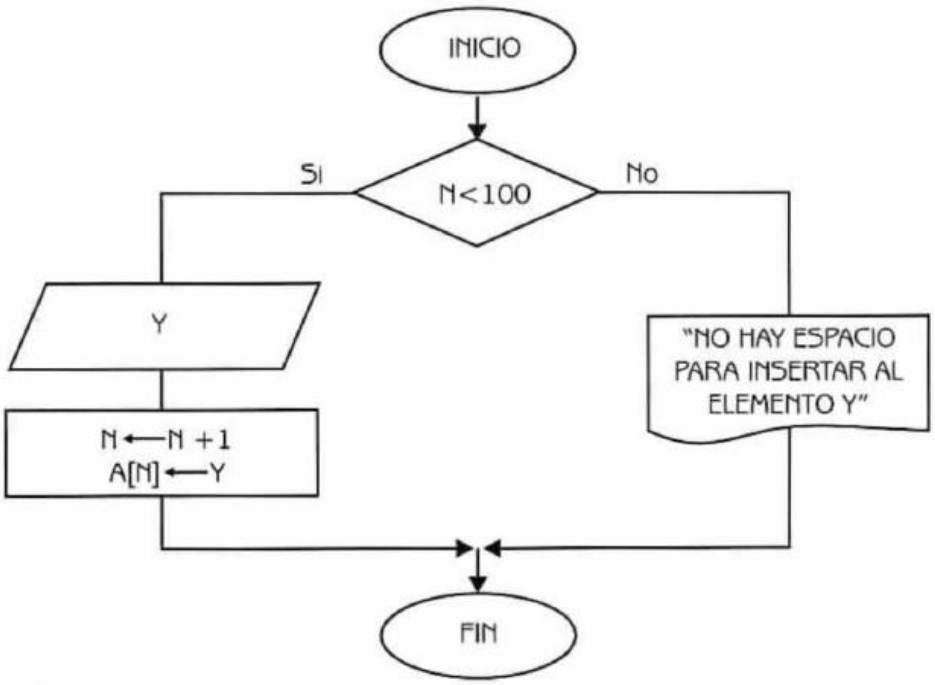


Diagrama de Flujo 4.3

A continuación presentamos el programa correspondiente.

Programa 4.3

```
INSERTA_DESORDENADO

{El programa inserta un elemento en un arreglo desordenado}.

{N y Y son variables de tipo entero. A es un arreglo unidimensional de tipo entero}

1. Si N < 100
        entonces
                Leer Y
                Hacer N ⟵ N + 1 y A[N] ⟵ Y
        sino
                Escribir "No hay espacio para insertar al elemento Y"
2. {Fin del condicional del paso 1}
```

Luego de la inserción, el arreglo A queda como se muestra en la figura 4.9.

www.FreeLibros.com

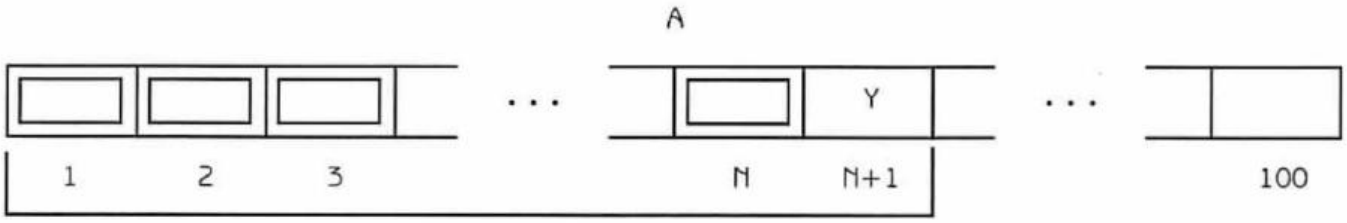


Figura 4.9 Inserción en arreglos desordenados.

a.2) Eliminación: Para eliminar un elemento X de un arreglo A desordenado debe verificarse que el arreglo no esté vacío y que X se encuentre en el arreglo. Si se cumplen estas condiciones entonces se procederá a recorrer todos los elementos que están a su derecha una posición a la izquierda, decrementando finalmente el número de componentes del arreglo. A continuación presentamos el diagrama de flujo correspondiente.

www.FreeLibros.com

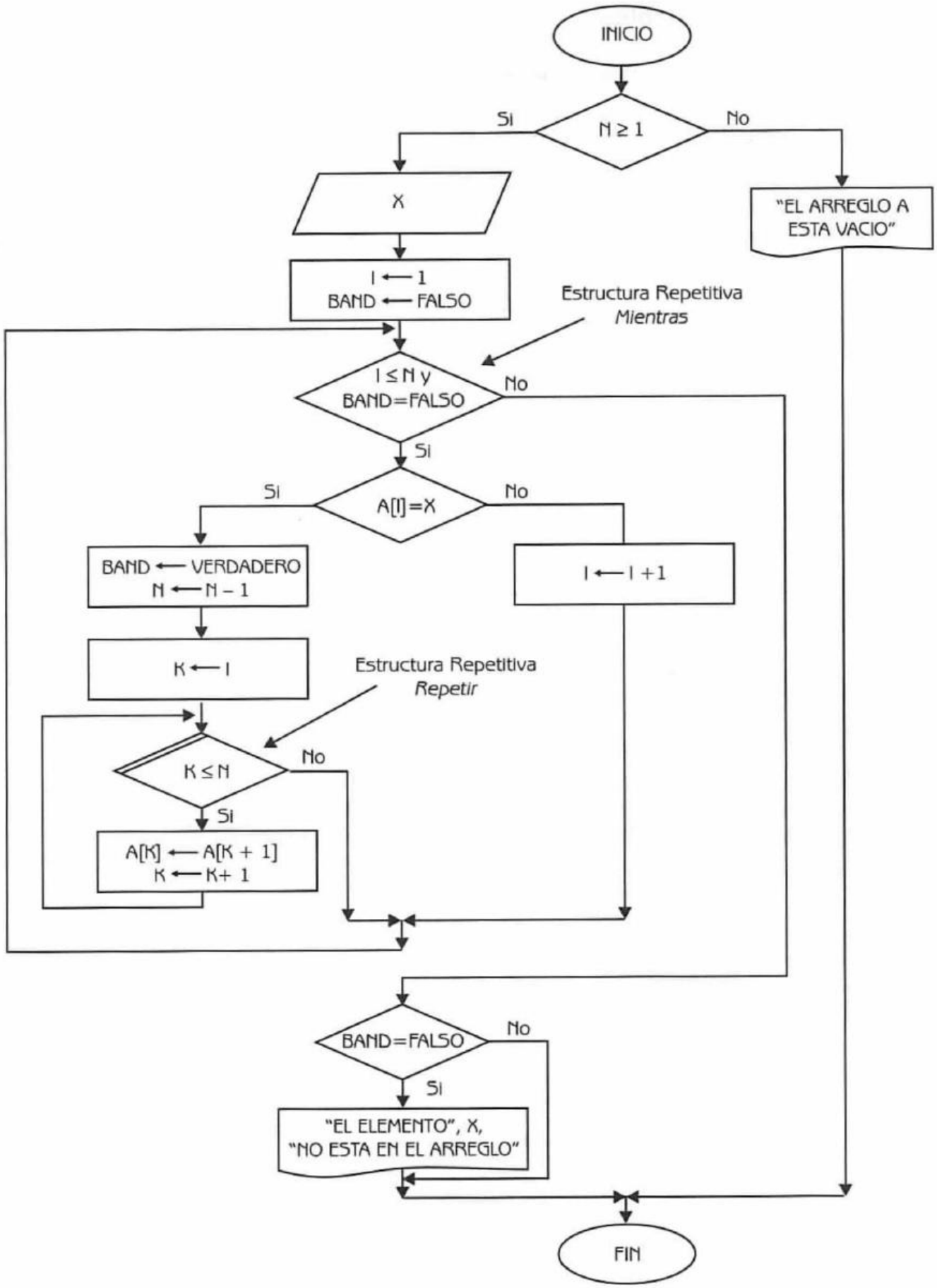


Diagrama de Flujo 4.4

www.FreeLibros.com

Explicación de las variables

N: Variable de tipo entero. Almacena el número actual de elementos del arreglo.

X: Variable de tipo entero. Representa al valor que se va a eliminar.

I: Variable de tipo entero. Se utiliza como variable de control del ciclo externo y como índice del arreglo V.

BAND: Variable de tipo booleano. Se inicializa en falso. Cambia su valor a verdadero si se encuentra el elemento a eliminar en el arreglo, en cuyo caso se interrumpe el ciclo.

K: Variable de tipo entero. Se utiliza como variable de control del ciclo interno y como índice del arreglo A.

A: Arreglo unidimensional de tipo entero. Su capacidad máxima es de 100 elementos.

A continuación presentamos el programa correspondiente.

Programa 4.4

```
ELIMINA DESORDENADO

{El programa elimina un elemento en un arreglo desordenado}

{N, X, I y K son variables de tipo entero. BAND es una variable de tipo booleano. A
es un arreglo unidimensional de tipo entero}

1. Si N >= 1
      entonces
         Leer X
         Hacer I ←— 1 y BAND ←— FALSO
   1.1 Mientras (I <= N) y (BAND = FALSO) Repetir
         1.1.1 Si A[I] = X
                  entonces
                     Hacer BAND ←— VERDADERO, N ←— N–1 y K ←— I
                  1.1.1.1 Repetir con K desde I hasta N
                                 Hacer A[K] ←— A[K+1] y K ←— K + 1
                  1.1.1.2 {Fin del ciclo del paso 1.1.1.1}
                  sino
                        Hacer I ←— I + 1
         1.1.2 {Fin del condicional del paso 1.1.1}
   1.2 {Fin del ciclo del paso 1.1}
```

www.FreeLibros.com

```
        1.3 Si BAND = FALSO entonces
                Escribir "El elemento",X, "no está en el arreglo"
        1.4 {Fin del condicional del paso 1.3}
        sino
            Escribir "El arreglo A está vacío"
2. {Fin del condicional del paso 1}
```

Luego de la eliminación, el arreglo A queda como se muestra en la figura 4.10.

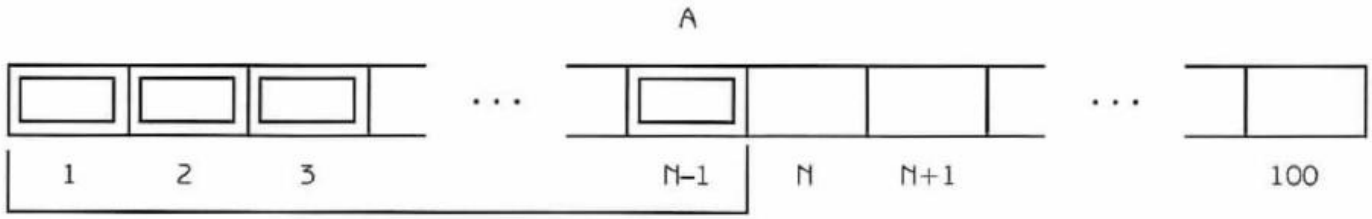


Figura 4.10 Eliminación en arreglos desordenados.

a.3) Modificación: Para modificar un elemento X por un elemento Y, de un arreglo A que se encuentra desordenado debe verificarse que el arreglo no esté vacío y que X se encuentre en el arreglo. Si se cumplen estas condiciones entonces se procederá a su actualización. Puede observarse que existen tareas comunes con la operación de eliminación:

- Determinar que el arreglo no esté vacío.
- Encontrar el elemento a modificar (eliminar).

A continuación presentamos el diagrama de flujo correspondiente.

www.FreeLibros.com

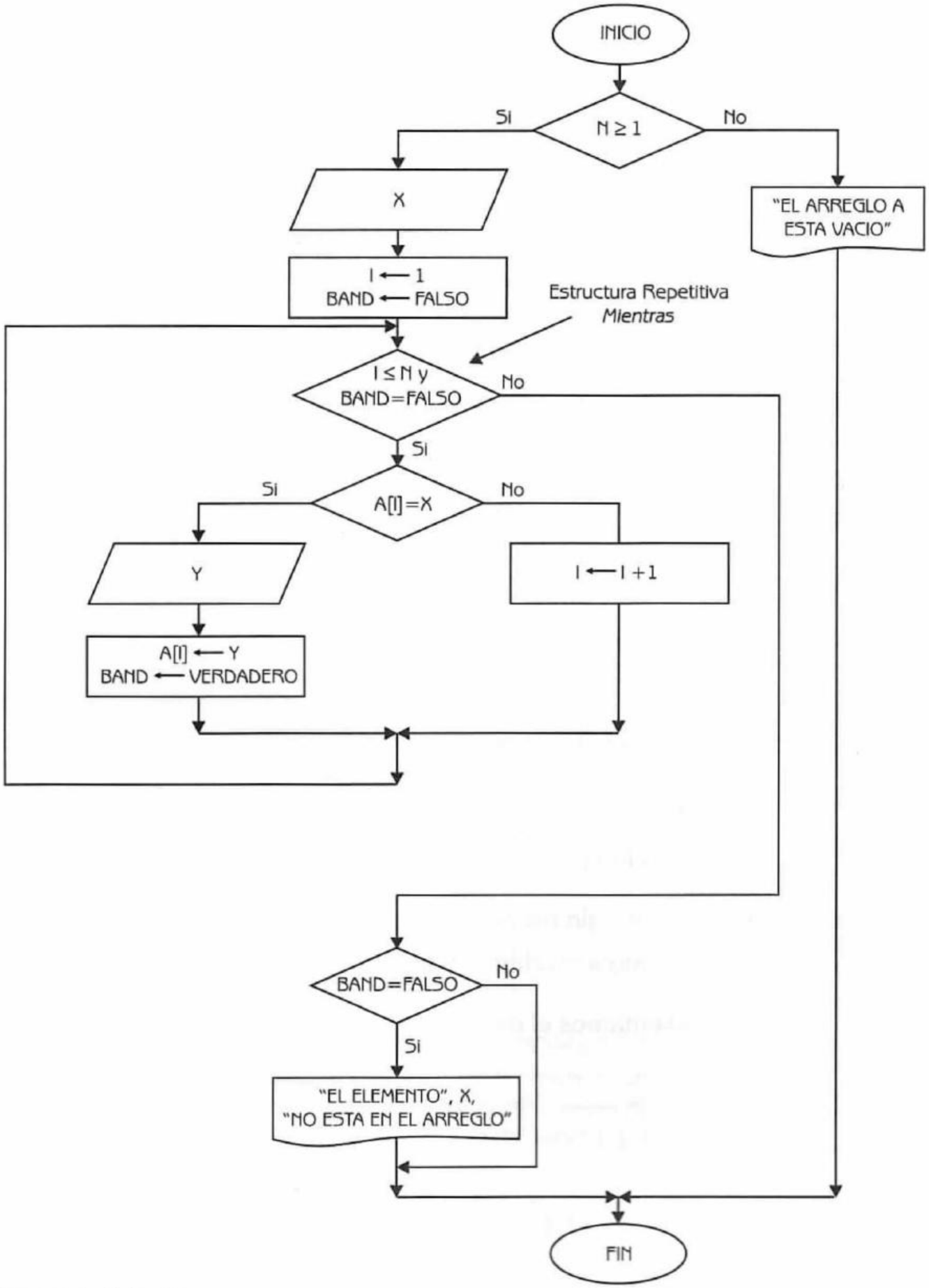


Diagrama de Flujo 4.5

www.FreeLibros.com

Explicación de las variables

N: Variable de tipo entero. Almacena el número actual de elementos del arreglo.
X: Variable de tipo entero. Representa el elemento que se va a modificar.
I: Variable de tipo entero. Se utiliza como variable de control del ciclo y como índice del arreglo.
BAND: Variable de tipo booleano. Se inicializa en FALSO. Cambia su valor a VERDADERO si se encuentra el elemento a modificar en el arreglo, en cuyo caso se interrumpe el ciclo.
A: Arreglo unidimensional de tipo entero. Su capacidad máxima es de 100 elementos.
Y: Variable de tipo entero. Representa al elemento que se introduce y modifica al elemento X.

A continuación presentamos al diagrama de flujo en lenguaje algorítmico.

Programa 4.5

```
MODIFICA DESORDENADO

{El programa modifica un elemento X por un elemento Y en un arreglo desordenado}

{N, X, I y Y son variables de tipo entero. BAND es una variable de tipo booleano. A es un arreglo unidimensional de tipo entero}

1. Si N >= 1
      entonces
            Leer X
            Hacer I ←— 1 y BAND ←— FALSO
      1.1 Mientras (I <= N) y (BAND = FALSO) Repetir
            1.1.1 Si A[I] = X
                        entonces
                              Leer Y
                              Hacer A[I] ←— Y y BAND ←— VERDADERO
                        sino
                              Hacer I ←— I + 1
            1.1.2 {Fin del condicional del paso 1.1.1}
      1.2 {Fin del ciclo del paso 1.1}
      1.3 Si BAND = FALSO entonces
```

www.FreeLibros.com

```
                Escribir "El elemento", X, "no está en el arreglo"
        1.4 {Fin del condicional del paso 13}
        sino
                Escribir "El arreglo A está vacío"
2. {Fin del condicional del paso 1}
```

Luego de la modificación, el arreglo A queda como se muestra en la figura 4.11.

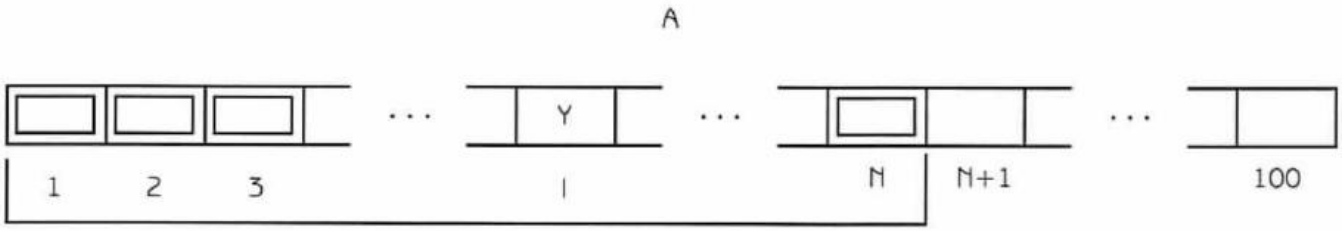


Figura 4.11 Modificación en arreglos desordenados.

b) ARREGLOS ORDENADOS

Considere el arreglo ordenado A de 100 elementos de la figura 4.12. Los primeros N componentes del mismo tienen asignado un valor. En este caso se trabajará con un arreglo ordenado de manera creciente, es decir:

$$A[1] \leq A[2] \leq A[3] \leq \ldots \leq A[N]$$

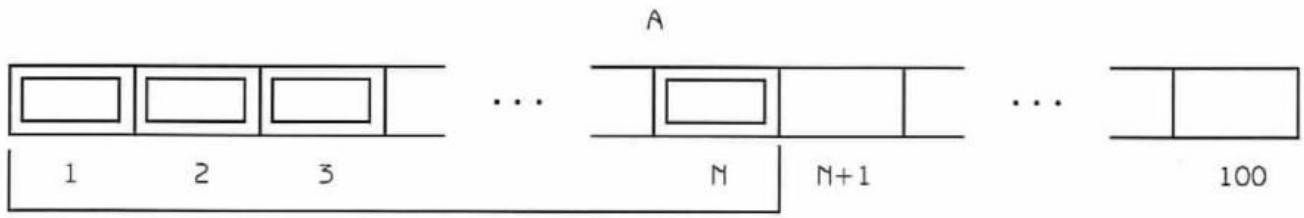


Figura 4.12 Actualización de arreglos ordenados.

Cuando se trabaja con arreglos ordenados no se debe alterar el orden al insertar nuevos elementos o al modificar los existentes.

b1) Inserción: Para insertar un elemento X en un arreglo A que se encuentra ordenado, debe verificarse que exista espacio. Luego tendrá que encontrarse la posición en la que debería estar el nuevo valor para no alterar el orden del arreglo. Una vez detectada la

www.FreeLibros.com

posición, se procederá a recorrer todos los elementos desde la misma hasta la N–ésima posición, un lugar a la derecha. Finalmente se asignará el valor de X en la posición encontrada (Los pasos del recorrimiento no se llevan a cabo cuando el valor a insertar es mayor que el último elemento del arreglo).

Generalmente cuando se quiere hacer una inserción debe verificarse que el elemento no se encuentre en el arreglo. En la mayoría de los casos prácticos no interesa tener información duplicada, por lo tanto si el valor a insertar ya estuviera en el arreglo, la operación no se llevaría a cabo.

Debemos señalar que tanto en los procesos de inserción y eliminación en arreglos ordenados es recomendable tener un procedimiento que busque el elemento X en el arreglo ordenado. Este procedimiento dará como resultado la posición en la que se encuentre el elemento X (en cuyo caso el elemento ya pertenece al arreglo y por lo tanto no debemos insertarlo) o el negativo de la posición en la que debería estar.

A continuación en el diagrama de flujo 4.6, presentamos la solución al problema.

www.FreeLibros.com

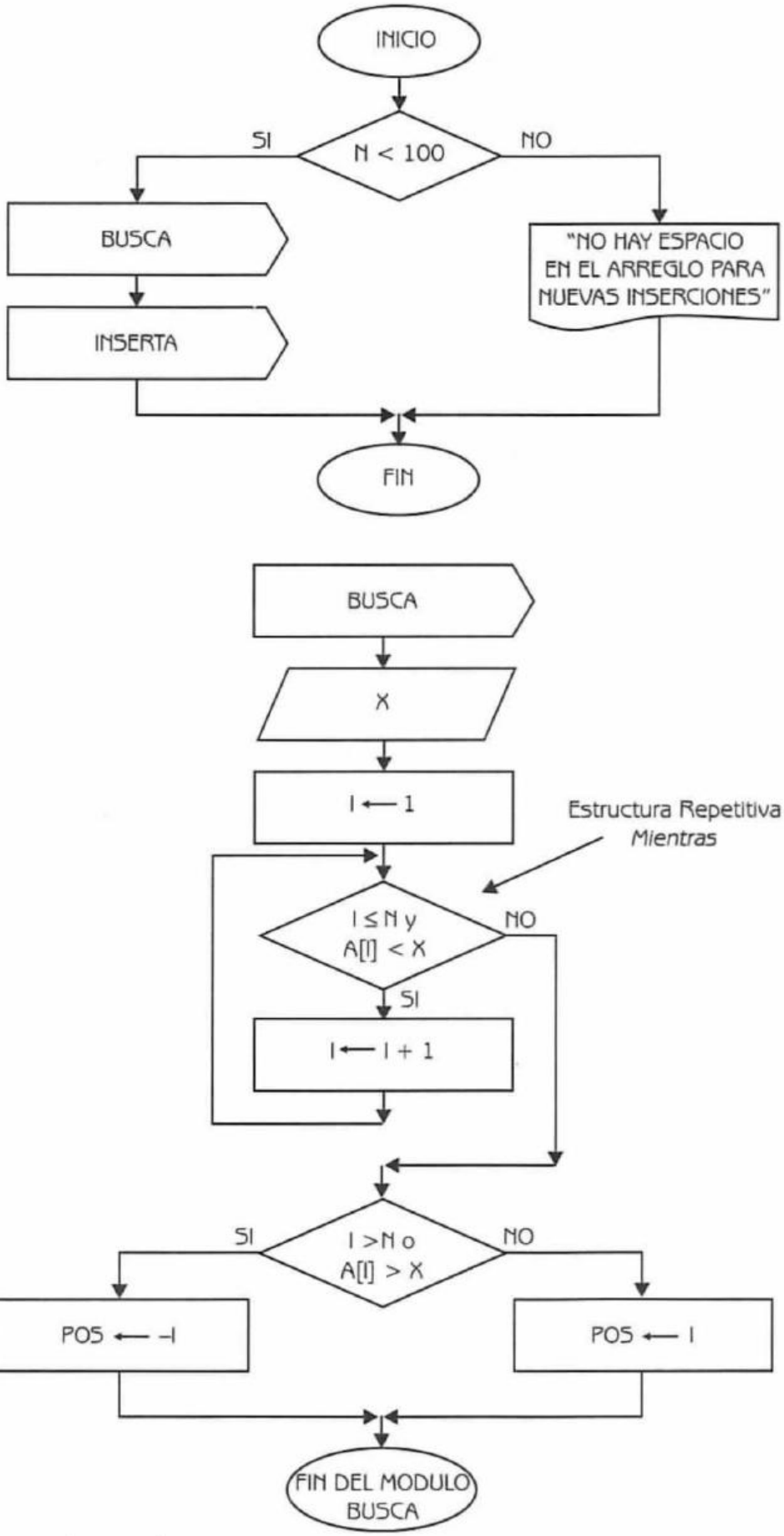


Diagrama de Flujo 4.6 (*continúa*)

www.FreeLibros.com

Nota: Observe el lector que si hay espacio para insertar, se modulariza el problema BUSCANDO primero si se encuentra el elemento y posteriormente, si es posible, insertándolo.

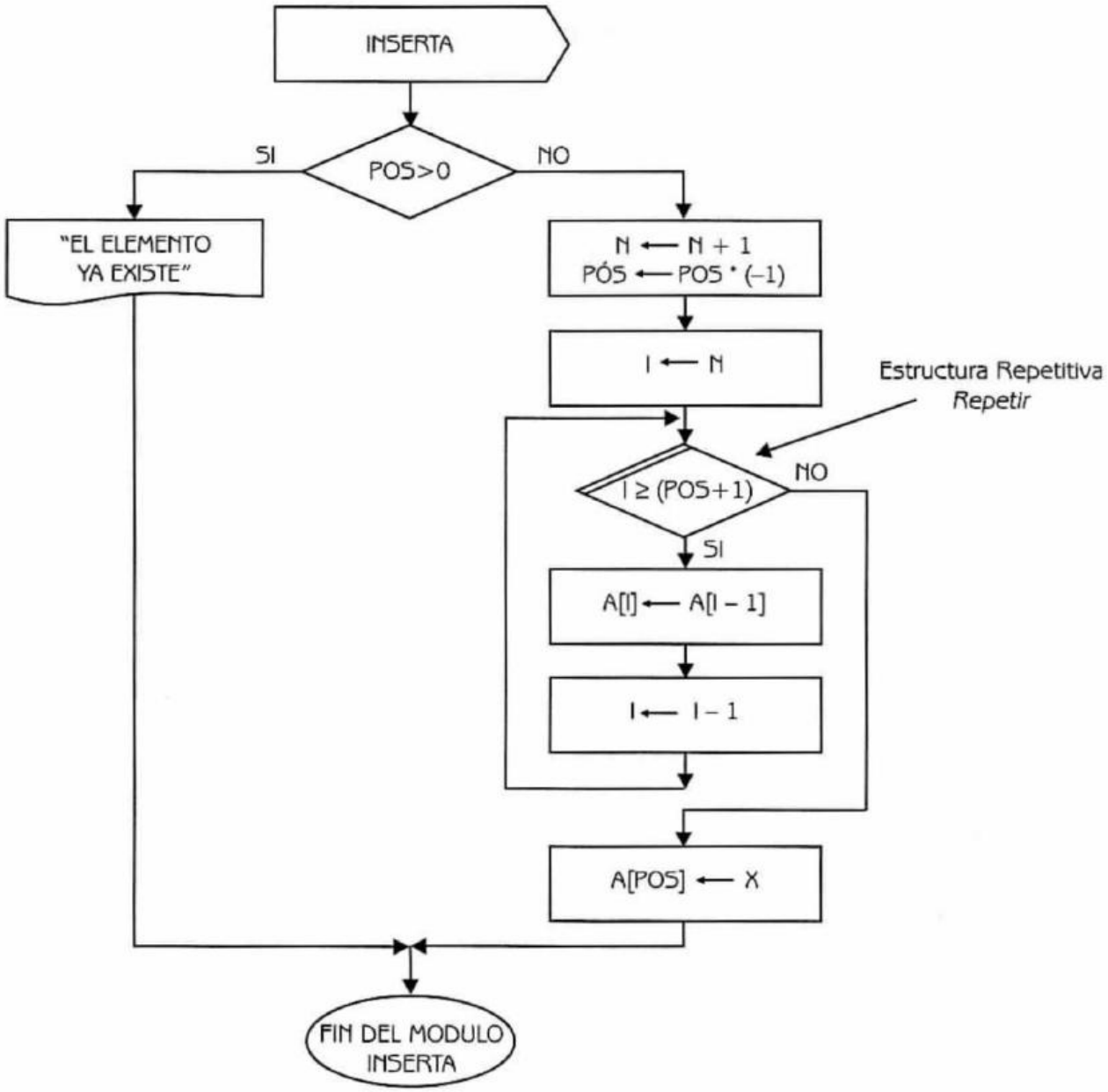


Diagrama de Flujo 4.6 (*continuación*)

Nota: En este módulo se inserta un elemento X en un arreglo ordenado A de N elementos.

Explicación de las variables

N: Variable de tipo entero. Almacena el número actual de elementos del arreglo.

www.FreeLibros.com

I: Variable de tipo entero. Se utiliza como variable de control de los ciclos. En el módulo BUSCA, detecta además la posición donde se encuentra o debería estar el elemento X. Es utilizado también como índice del arreglo.

X: Variable de tipo entero. Representa al elemento que se va a insertar (si no se encuentra en el arreglo).

A: Arreglo unidimensional de tipo entero. Su capacidad máxima es de 100 elementos.

POS: Variable de tipo entero. Almacena la posición donde se encuentra o debería estar el elemento X.

A continuación presentamos el diagrama de flujo 4.6 en lenguaje algorítmico.

Programa 4.6

```
INSERTA_ORDENADO

El programa, si es posible, inserta un elemento X en un arreglo ordenado A de N
elementos

N, I, X y POS son variables de tipo entero. A es un arreglo unidimensional de tipo
entero

1. Si N < 100
      entonces
            Leer X
            Hacer I ←—— 1
   1.1 Mientras (I <= N) y (A[I] < X) Repetir
            Hacer I ←—— I + 1
   1.2 Fin del ciclo del paso 1.1
   1.3 Si (I > N) ó (A[I] > X)
         entonces
            Hacer POS ←—— -I
         sino
            Hacer POS ←—— I
   1.4 {Fin del condicional del paso 1.3}
   1.5 Si POS > 0
         entonces
               Escribir "El elemento ya existe"
         sino
               Hacer N ←—— N + 1, POS ←—— POS * (-1)
```

www.FreeLibros.com

```
                    e I ←—— N
          1.5.1 Repetir con I desde N hasta (POS + 1)
                    Hacer A[I] ←—— A[I – 1] e I ←—— I – 1
          1.5.2 {Fin del ciclo del paso 1.5.1}
                    Hacer A[POS] ←—— X
    1.6 {Fin del condicional del paso 1.5}
    sino
          Escribir "No hay espacio en el arreglo para nuevas inserciones"
2. {Fin del condicional del paso 1}
```

Luego de la inserción el arreglo queda como se muestra en la figura 4.13.

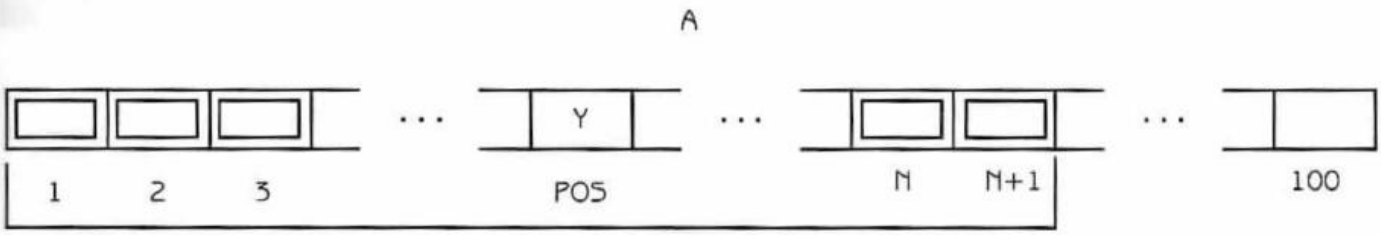
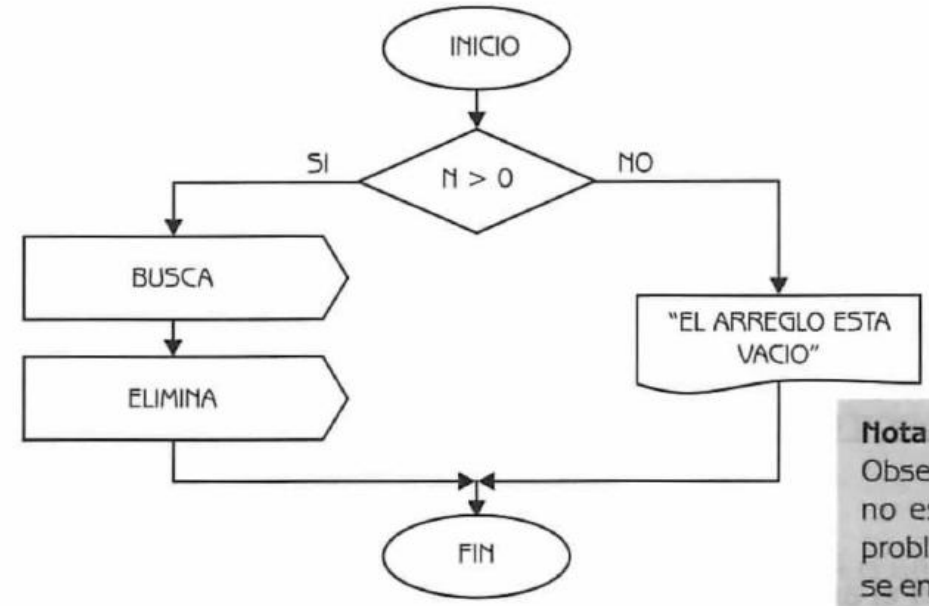


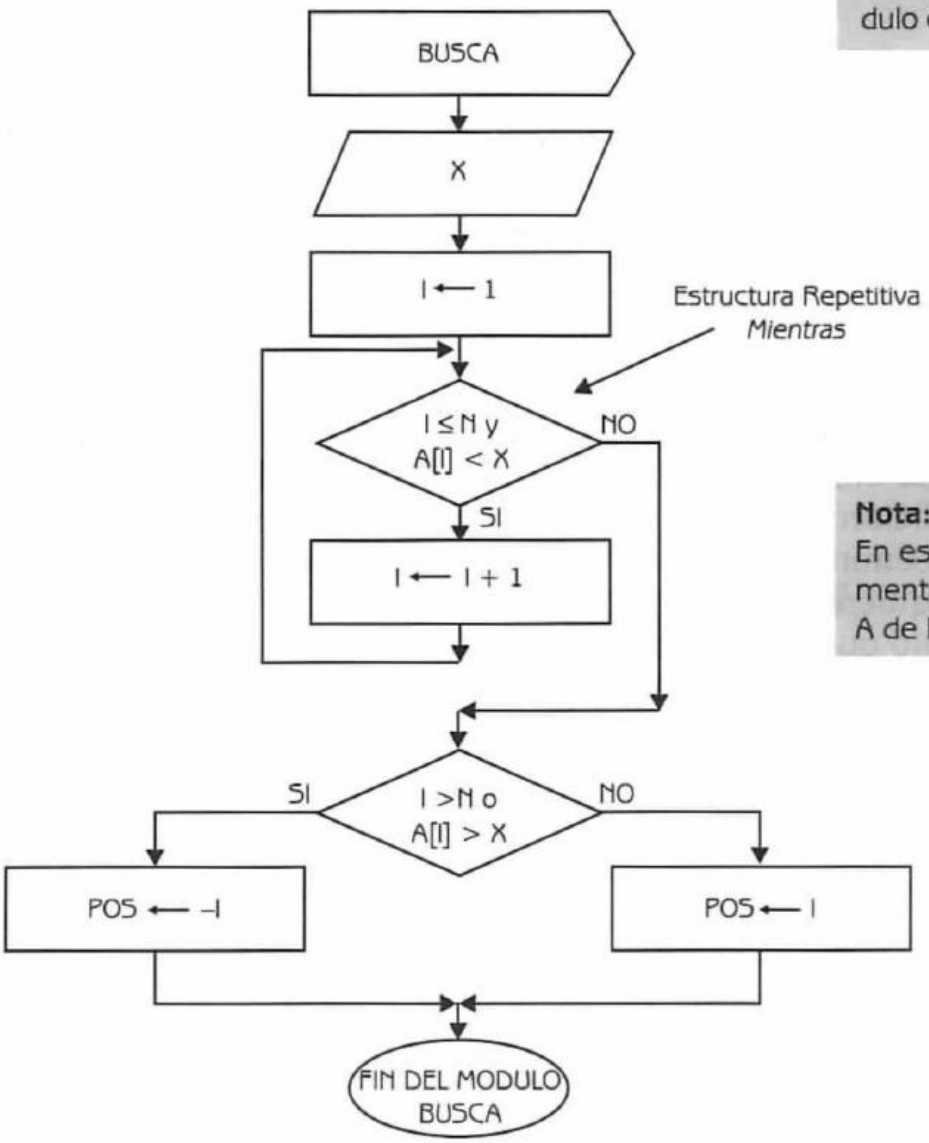
Figura 4.13 Inserción en arreglos ordenados.

b.2) Eliminación: Para eliminar un elemento X de un arreglo ordenado A, debe verificarse que el arreglo no esté vacío. Si se cumple esta condición, entonces tendrá que buscarse la posición del elemento a eliminar. Si el resultado del módulo BUSCA es un valor positivo, quiere decir que el elemento se encuentra en el arreglo y por lo tanto puede ser eliminado; en otro caso no se puede ejecutar la operación. A continuación en el diagrama de flujo 4.7 presentamos la solución al problema.

www.FreeLibros.com



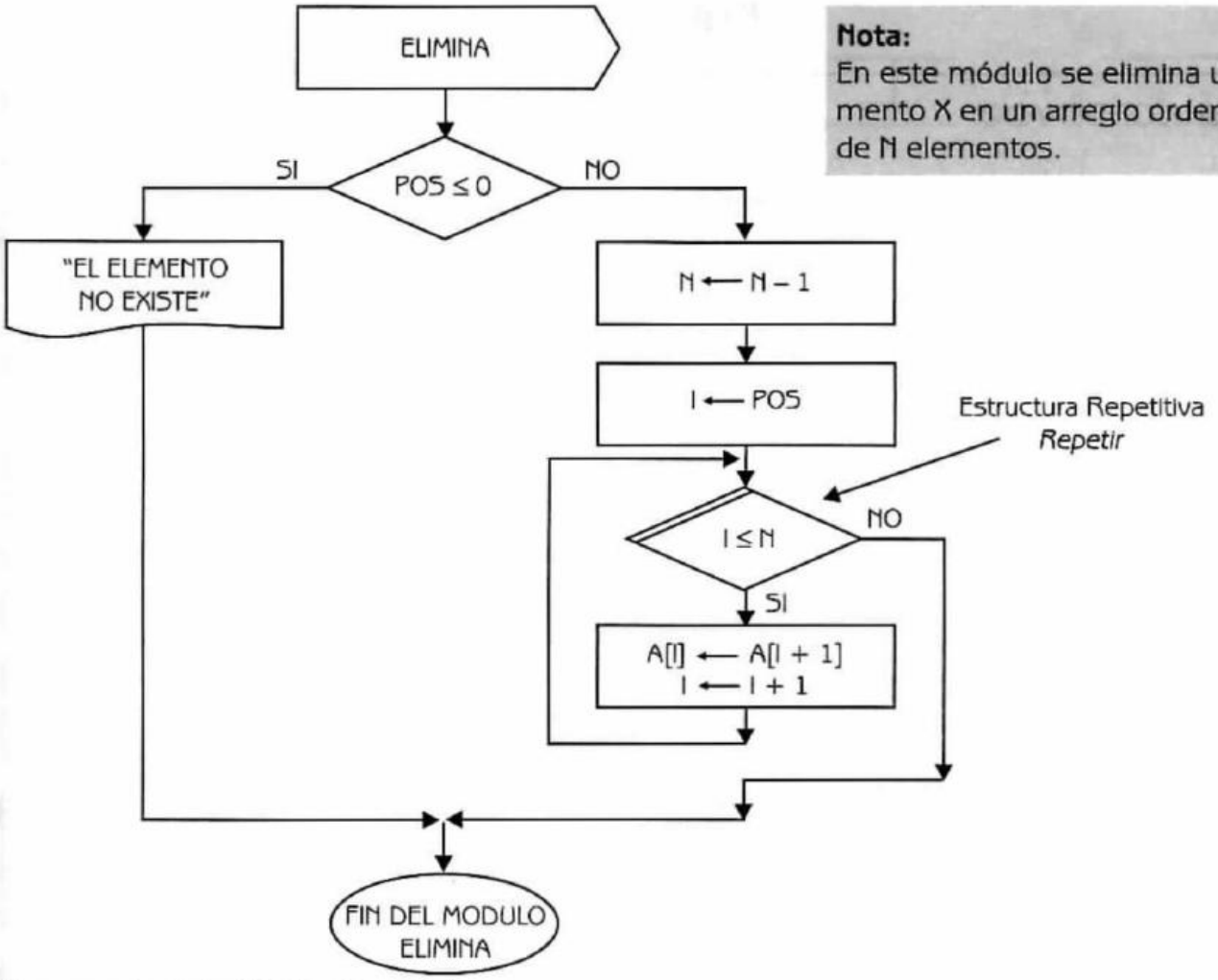
Nota:
Observe el lector que si el arreglo no está vacío, se modulariza el problema BUSCANDO primero si se encuentra el elemento y posteriormente, si el resultado del módulo es positivo, eliminándolo.



Nota:
En este módulo se busca un elemento X en un arreglo ordenado A de N elementos.

Diagrama de Flujo 4.7 (*continúa*)

www.FreeLibros.com



Nota:
En este módulo se elimina un elemento X en un arreglo ordenado A de N elementos.

Diagrama de Flujo 4.7 (*continuación*)

Explicación de las variables

N: Variable de tipo entero. Almacena el número actual de elementos del arreglo.

I: Variable de tipo entero. Se utiliza como variable de control de los ciclos. En el módulo BUSCA almacena además la posición donde se encuentra o debería estar el elemento X. Es utilizado también como índice del arreglo.

X: Variable de tipo entero. Representa al elemento que se va a eliminar (si se encuentra en el arreglo).

A: Arreglo unidimensional de tipo entero. Su capacidad máxima es de 100 elementos.

POS: Variable de tipo entero. Almacena la posición donde se encuentra o debería estar el elemento X.

A continuación presentamos el diagrama de flujo 4.7 en lenguaje algorítmico.

www.FreeLibros.com

Programa 4.7

```
ELIMINA ORDENADO

{El programa, si es posible, elimina un elemento X de un arreglo ordenado A de N
elementos}

{N, I, X y POS son variables de tipo entero. A es un arreglo unidimensional de tipo
entero}

1. Si N > 0
      entonces
            Leer X
            Hacer I ←—— 1
      1.1 Mientras (I <= N) y (A[I] < X) Repetir
                  Hacer I ←—— I + 1
      1.2 {Fin del ciclo del paso 1.1}
      1.3 Si (I > N) o (A[I] > X)
            entonces
                  Hacer POS ←—— -I
            sino
                  Hacer POS ←—— I
      1.4 {Fin del condicional del paso 1.3}
      1.5 Si POS <= 0
            entonces
                  Escribir "El elemento no existe"
            sino
                  Hacer N ←—— N-1 e I ←—— POS
            1.5.1 Repetir con I desde POS hasta N
                        Hacer A[I] ←—— A[I + 1] e I ←—— I + 1
            1.5.2 {Fin del ciclo del paso 1.5.1}
      1.6 {Fin del condicional del paso 1.5}
      sino
          Escribir "El arreglo está vacío"
2. {Fin del condicional del paso 1}
```

Luego de la eliminación, el arreglo queda como se muestra en la figura 4.14.

www.FreeLibros.com

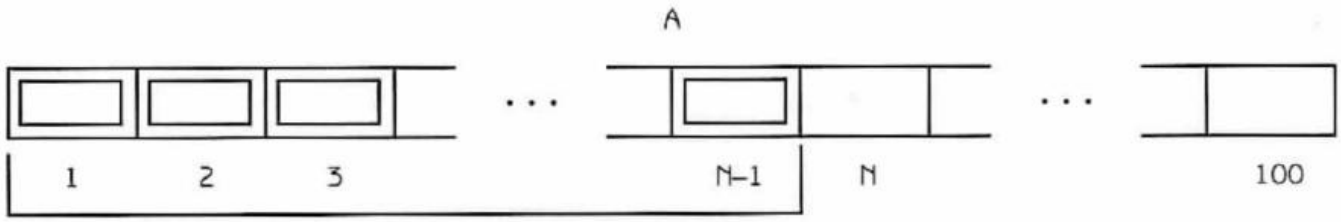


Figura 4.14 Eliminación en arreglos ordenados.

b.3) Modificación: se procede de manera similar a la eliminación de un elemento en un arreglo ordenado. La variante se presenta en que al modificar el valor X por un valor Y, debe verificarse que el orden del arreglo no se altere. Si esto llegara a suceder, entonces debería reordenarse el arreglo, o bien eliminar primero el elemento a modificar e insertar posteriormente el nuevo elemento.

Hasta el momento hemos visto tanto la forma de declarar arreglos como de usarlos. Ahora podemos darle solución al problema del ejemplo 4.1.

Ejemplo 4.5

Procederemos entonces a darle solución al problema del ejemplo 4.1 utilizando tipos estructurados de datos.

Datos: $S_1, S_2, S_3, \ldots, S_{70}$

Donde:

S_i es una variable de tipo real que representa al sueldo del empleado i $(1 \leq i \leq 70)$.

www.FreeLibros.com

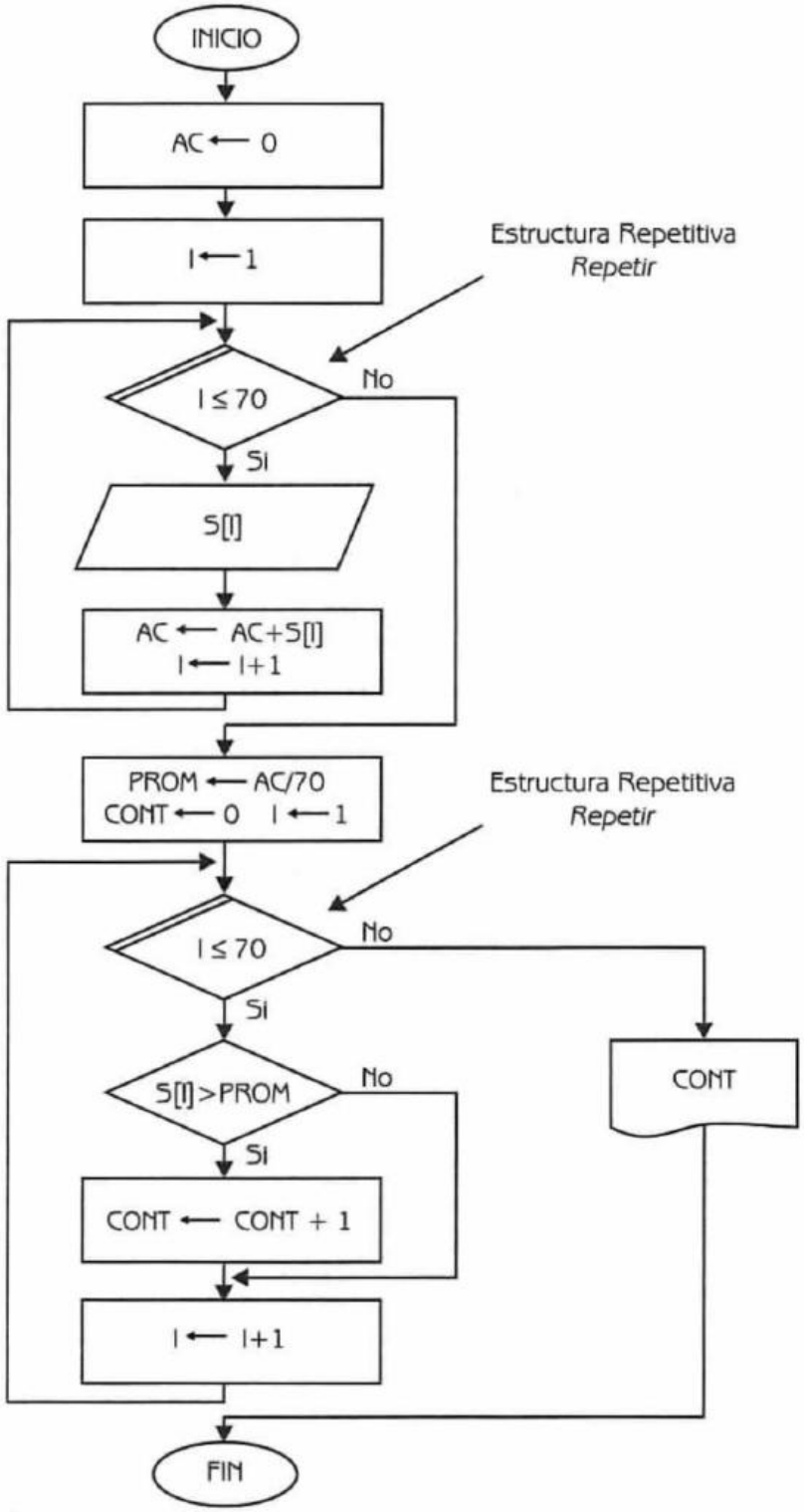


Diagrama de Flujo 4.8

Nota: Observe el lector que con esta solución se evitan los problemas mencionados anteriormente. Se realiza una única lectura de los datos y además se define una única variable para almacenar los 70 sueldos. Al usar el arreglo puede disponerse de los datos tantas veces como sea necesario sin tener que volver a leerlos, ya que éstos permanecen en memoria. Además se facilita el procesamiento de los datos, al poder generalizar ciertas operaciones.

www.FreeLibros.com

Explicación de las Variables

AC: Variable de tipo entero. Acumula los sueldos de los empleados.
I: Variable de tipo entero. Se utiliza como variable de control de los ciclos. Es utilizada también como índice del arreglo.
S: Arreglo unidimensional de tipo real. Almacena los sueldos de los 70 empleados.
PROM: Variable de tipo real. Almacena el promedio de los sueldos de los 70 empleados de la empresa.
CONT: Variable de tipo entero. Cuenta los empleados que tienen un sueldo mayor al promedio.

A continuación presentamos el diagrama de flujo en lenguaje algorítmico.

Programa 4.8

```
CON_ARREGLOS

{El programa resuelve el problema del ejemplo 4.1 aplicando arreglos}

{AC, I y CONT son variables de tipo entero. PROM es una variable de tipo real. S es
un arreglo unidimensional de tipo real}

1. Hacer AC ←— 0 e I ←— 1
2. Repetir con I desde 1 hasta 70
        Leer S[I]
        Hacer AC ←— AC + S[I] e I ←— I + 1
3. {Fin del ciclo del paso 2}
4. Hacer PROM ←— AC/70, CONT ←— 0 e I ←— 1
5. Repetir con I desde 1 hasta 70
   5.1 Si S[I] > PROM entonces
           Hacer CONT ←— CONT + 1
   5.2 {Fin del condicional del paso 5.1}
       Hacer I ←— I + 1
6. {Fin del ciclo del paso 5}
7. Escribir CONT
```

4.3 Arreglos Multidimensionales

Los arreglos presentados hasta el momento reciben el nombre de arreglos unidimensionales o lineales, debido a que cualquier elemento es referenciado por un único índice. También existe en la mayoría de los lenguajes de programación

www.FreeLibros.com

una estructura de arreglos multidimensionales. El número de dimensiones (índices) que se deben de utilizar en un arreglo depende del problema que debemos resolver y las características del lenguaje que utilicemos.

Se analizarán primero los arreglos bidimensionales (caso especial de los multidimensionales) por ser los más utilizados.

4.3.1 Arreglos Bidimensionales

Para comprender mejor la estructura de los arreglos bidimensionales, presentamos a continuación un ejemplo.

Ejemplo 4.6

La tabla 4.2 contiene las lluvias caídas medidas en milímetros, correspondientes a los 12 meses del año anterior en cuatro estados de la República Mexicana.

Tabla 4.2 Lluvias mensuales por estados

Meses / Lluvias	Morelos	México	Querétaro	Puebla
Enero	50	45	60	58
Febrero	7	3	15	22
Marzo	12	10	8	17
Abril	15	5	20	35
Mayo	22	30	15	22
Junio	50	90	60	100
Julio	85	130	20	88
Agosto	70	75	88	94
Septiembre	65	49	53	105
Octubre	28	37	29	37
Noviembre	35	15	22	4
Diciembre	17	8	14	0

La tabla se interpreta de la siguiente manera: dado un mes, se conocen las lluvias caídas en cada uno de los estados; y dado un estado, se conocen las lluvias caídas en forma mensual. Si se quisiera almacenar esta información con los arreglos que se conocen, se tendrían dos alternativas:

1. Definir 12 arreglos de 4 elementos cada uno. Cada arreglo almacenará la información relativa a un mes.

www.FreeLibros.com

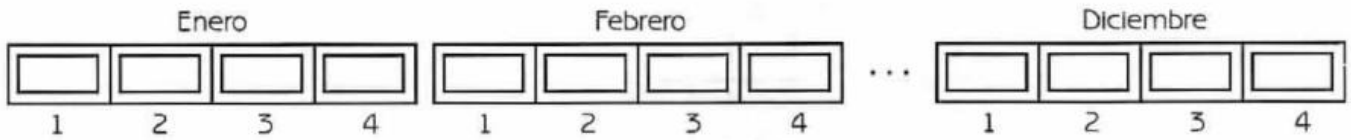


Figura 4.15 Almacenamiento de la información por mes.

2. Definir 4 arreglos de 12 elementos cada uno. Cada arreglo almacenará la información relativa a un estado a lo largo del año.

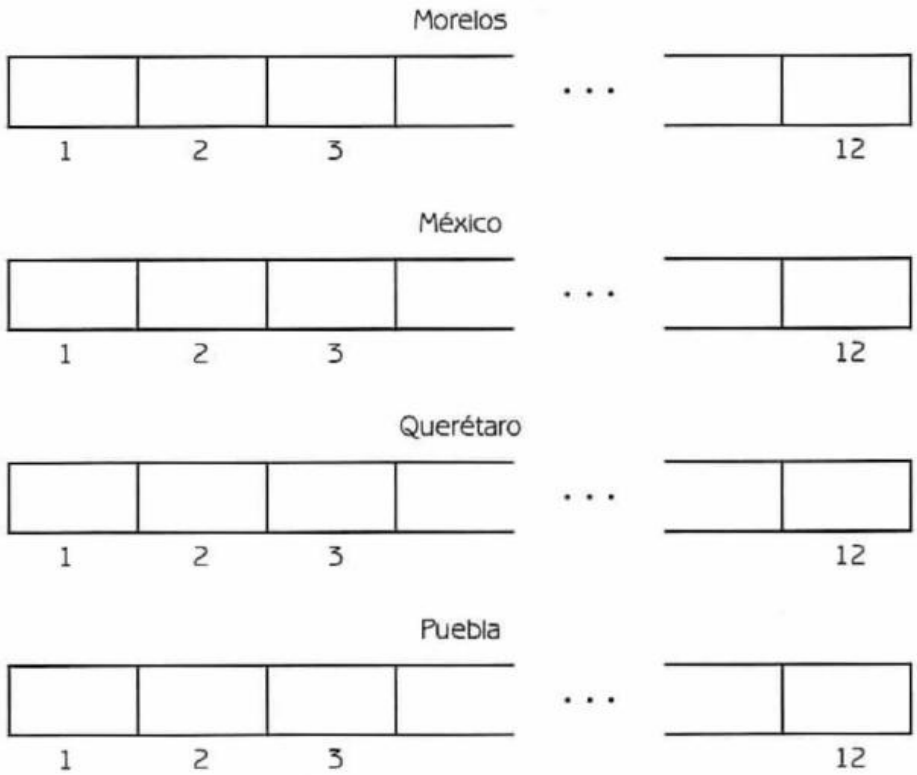


Figura 4.16 Almacenamiento de la información por estado.

Sin embargo, adoptar alguna de las dos alternativas no resulta muy práctico. Se necesita una estructura que permita manejar los datos considerando los meses (renglones de la tabla) y los estados (columnas de la tabla). Es decir, una estructura que trate a la información como un todo. La estructura que tiene esta característica se denomina *arreglo bidimensional*.

Un *arreglo bidimensional* es un conjunto de datos homogéneo, finito y ordenado, donde se hace referencia a cada elemento por medio de dos índices. El primero de los índices se utiliza para indicar el renglón y el segundo para indicar la columna. Un arreglo bidimensional también puede definirse como un arreglo de arreglos. En la figura 4.17 se presenta gráficamente un arreglo bidimensional.

El arreglo A(M × N) tiene M renglones y N columnas. Un elemento A[I,J] estará en el renglón I y en la columna J. Internamente en memoria se reservan M × N posiciones consecutivas para almacenar todos los elementos del arreglo.

www.FreeLibros.com

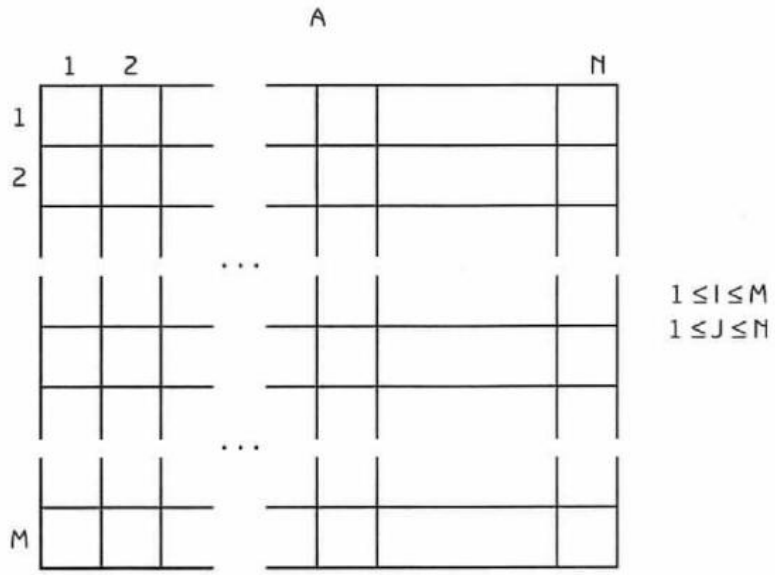


Figura 4.17 Representación de un arreglo bidimensional.

Declaración de arreglos bidimensionales

Se declararán los arreglos bidimensionales especificando el número de renglones y el número de columnas, junto con el tipo de los componentes.

id_arreglo = ARREGLO [líminfr..límisupr,líminfc..límsupc] DE tipo

Con **líminfr** y **límsupr** se declara el tipo del índice de los renglones y cuántos renglones tendrá el arreglo. Con **líminfc** y **límsupc** se declara el tipo de índice de las columnas y cuántas columnas tendrá el arreglo. Con tipo se declara el tipo de datos de todos los componentes del arreglo.

El número total de elementos (NTE) está determinado por la expresión:

$$NTE = (límsupr - líminfr + 1) * (límsupc - líminfc + 1)$$

Fórmula 4.2

Lo mismo que en el caso de los arreglos unidimensionales, los índices pueden ser cualquier tipo de dato ordinal (escalar, entero, caracter), mientras que los componentes pueden ser de cualquier tipo (reales, enteros, cadenas de caracteres, etc). Veamos a continuación algunos ejemplos de arreglos bidimensionales.

Ejemplo 4.7

Sea MATRIZ un arreglo bidimensional de números reales con índices enteros. Su representación queda como se muestra en la figura 4.18.

www.FreeLibros.com

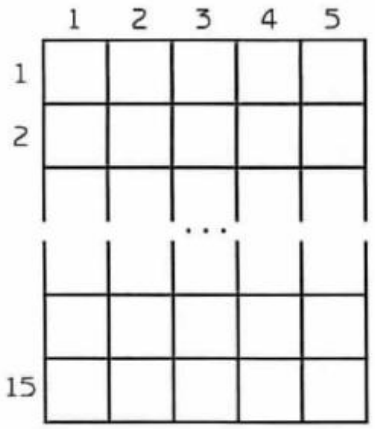


Figura 4.18

MATRIZ = ARREGLO [1..15,1..5] DE reales

- NTE = (15 –1 + 1) * (5 –1 + 1) = 15 * 5 = 75
- Cada elemento de MATRIZ será un número real. Para hacer referencia a cada uno de ellos se usarán dos índices y el nombre de la variable tipo arreglo: MATRIZ [i,j].

Donde: $1 \leq i \leq 15$
$1 \leq j \leq 5$

Así por ejemplo:

MATRIZ [2,4] hace referencia al elemento del renglón 2 y la columna 4.

MATRIZ [9,2] hace referencia al elemento del renglón 9 y la columna 2.

. . .

MATRIZ [15,5] hace referencia al elemento del renglón 15 y la columna 5.

Ejemplo 4.8

Consideremos el arreglo de reales del ejemplo anterior, utilicemos ahora índices negativos. Su representación queda como se muestra en la figura 4.19.

MATRIZ1 = ARREGLO [–11..3, –17.. –13] DE reales.

- NTE = (3 –(–11) + 1) * (–13 –(–17) + 1) = 15 * 5 = 75
- Cada elemento de MATRIZ1 será un número real. Para hacer referencia a cada uno de ellos se usarán dos índices y el hombre de la variable tipo arreglo: MATRIZ1 [i, j].

www.FreeLibros.com

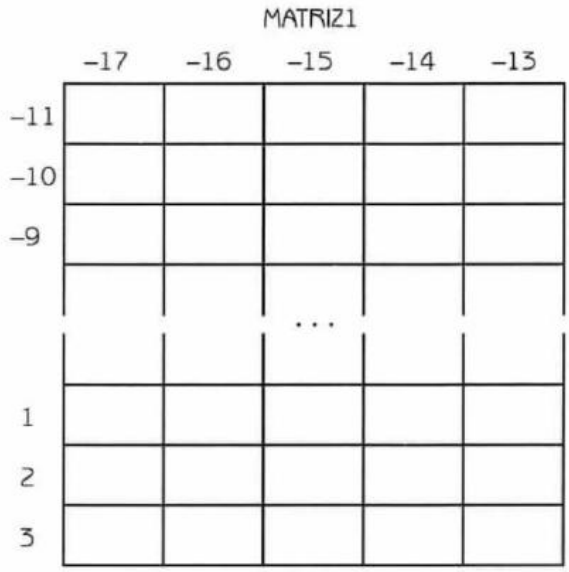


Figura 4.19

Donde: $-11 \leq i \leq 3$
$-17 \leq j \leq -13$

Así por ejemplo:

MATRIZ1 [–5, –16] hace referencia al elemento de la séptima fila y segunda columna.
MATRIZ1 [–4, –14] hace referencia al elemento de la octava fila y cuarta columna.
. . .
MATRIZ1 [3, –13] hace referencia al elemento de la última fila y última columna.

Ejemplo 4.9

Sea CADENA un arreglo bidimensional de tipo cadenas de caracteres con índices para los renglones de tipo caracter y para las columnas de tipo entero. Su representación queda como se muestra en la figura 4.20.

CADENA = ARREGLO ['a'.. 'z', –5..7] DE cadena de caracteres

- NTE = (ord('z') –ord('a') + 1) * (7 –(–5) + 1) =
  = (122 –97 + 1) * (7+ 5 + 1) = 26 * 13 = 338
- Cada elemento de CADENA será un valor tipo cadena de caracteres. Para hacer referencia a cada uno de ellos se usarán dos índices y el nombre de la variable tipo arreglo: CADENA [i, j].

www.FreeLibros.com

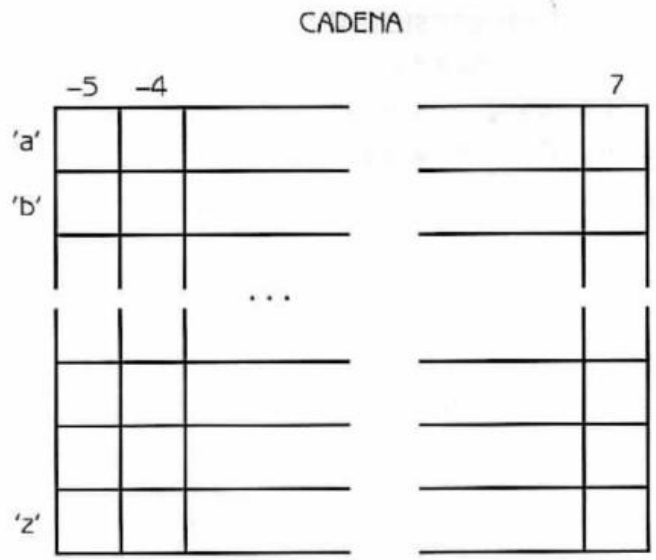


Figura 4.20

Donde: 'a' ≤ i ≤ 'z'

–5 ≤ j ≤ 7

Así por ejemplo:

CADENA ['b' , –3] hace referencia al elemento del segundo renglón y tercera columna.

CADENA ['e', 4] hace referencia al elemento de la quinta fila y décima columna.

. . .

CADENA ['z', 7] hace referencia al elemento de la última fila y última columna.

Operaciones con arreglos bidimensionales

A continuación se presentan las operaciones que pueden realizarse con arreglos bidimensionales:

- Lectura / Escritura.
- Asignación.
- Actualización: Inserción.
  Eliminación.
  Modificación.
- Ordenación.
- Búsqueda.

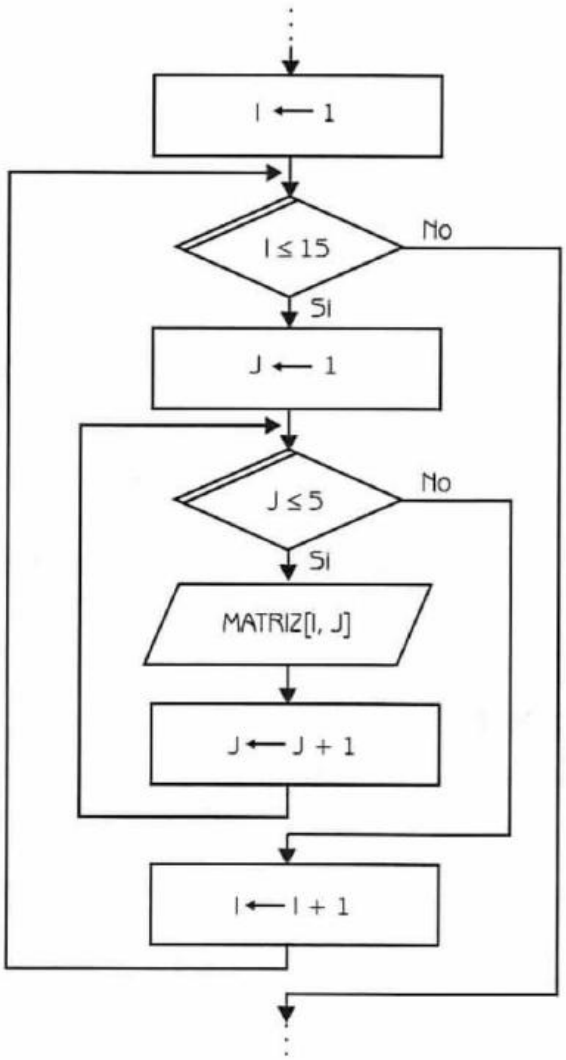
www.FreeLibros.com

En general los arreglos bidimensionales son una generalización de los unidimensionales, por lo que sólo presentaremos una rápida revisión de las operaciones de lectura, escritura y asignación. Para ilustrarlas utilizaremos el arreglo bidimensional MATRIZ presentado en el ejemplo 4.7.

Lectura

Cuando se introdujo la lectura en arreglos unidimensionales se dijo que se iban asignando valores a cada uno de los componentes. Lo mismo sucede con los arreglos bidimensionales. Sin embargo, como sus elementos deben referenciarse con dos índices, se deben utilizar dos ciclos para lograr la lectura de elementos consecutivos.

Supóngase que se desea leer todos los elementos del arreglo bidimensional MATRIZ. Los pasos a seguir son los siguientes:



www.FreeLibros.com

```
.
.
.
Hacer I ←—— 1
Repetir con I desde 1 hasta 15
        Hacer J ←—— 1
        Repetir con J desde 1 hasta 5
                Leer MATRIZ [I, J]
                Hacer J ←—— J + 1
        {Fin del ciclo interno}
        Hacer I ←—— I + 1
{Fin del ciclo externo}
.
.
.
```

Al variar los índices de I y J se lee un elemento de matriz, según la posición indicada por los índices I y J.

Para I = 1 y J = 1, se lee el elemento del renglón 1 y columna 1.

I = 1 y J = 2, se lee el elemento del renglón 1 y columna 2.

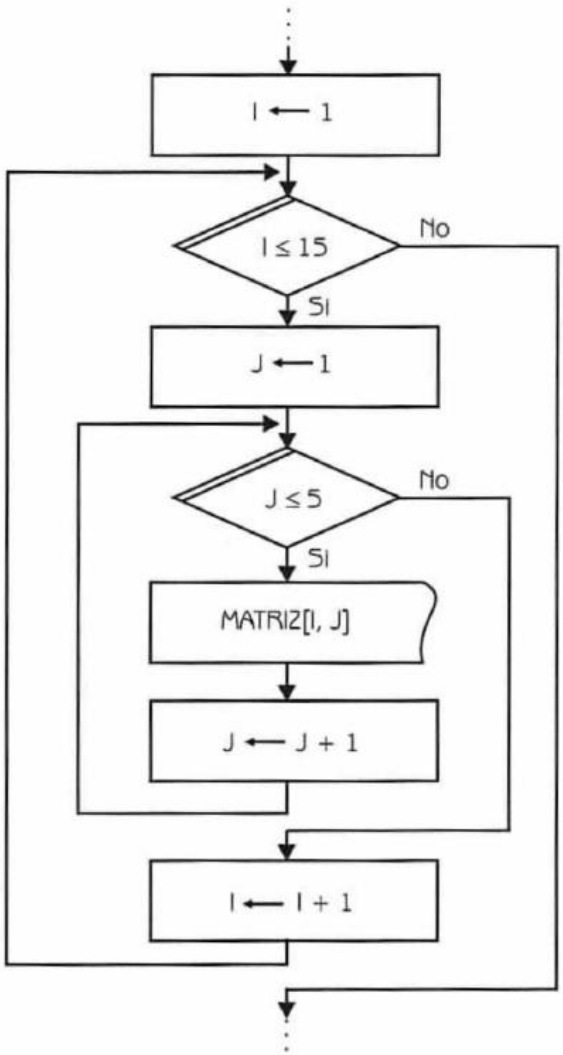
. . .

I = 15 y J = 5, se lee el elemento del renglón 15 y columna 5.

Escritura

La escritura de un arreglo bidimensional también se lleva a cabo elemento por elemento. Supóngase que se quiere escribir todos los componentes del arreglo MATRIZ. Los pasos a seguir son los que se muestran a continuación:

www.FreeLibros.com



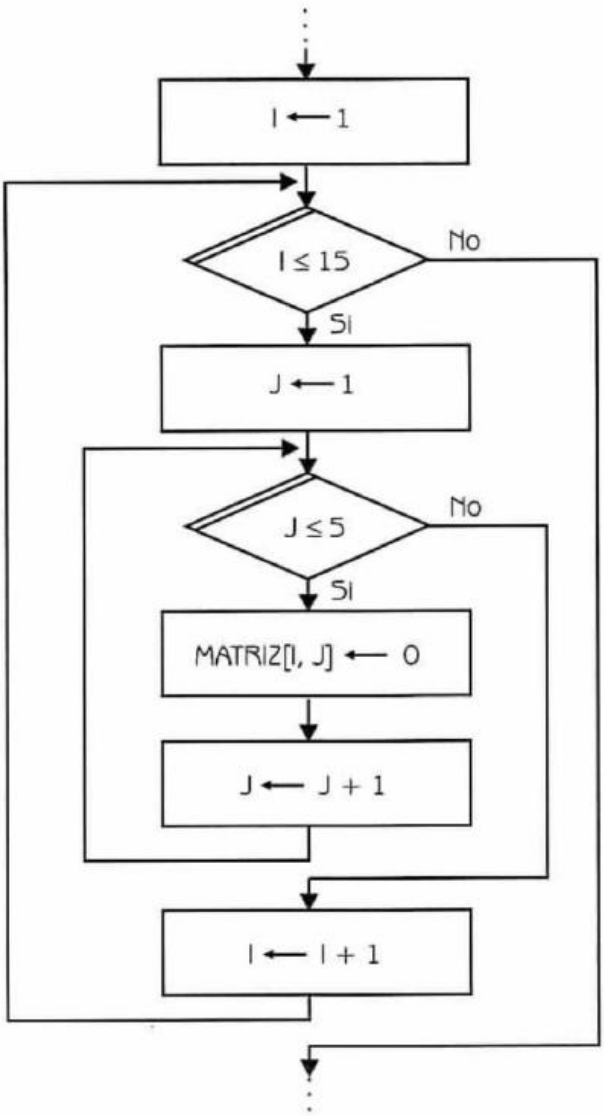

```
.
.
.
Hacer I ←——— 1
Repetir con I desde 1 hasta 15
        Hacer J ←——— 1
        Repetir con J desde 1 hasta 5
                Escribir MATRIZ [I, J]
                Hacer J ←——— J + 1
        {Fin del ciclo interno}
        Hacer I ←——— I + 1
{Fin del ciclo externo}
.
.
.
```

Asignación

La asignación de valores a un arreglo bidimensional puede realizarse de dos maneras diferentes, según el número de componentes involucrados.

www.FreeLibros.com

1. *A todos los elementos del arreglo*: en este caso se necesitarán dos ciclos para recorrer todo el arreglo.




```
.
.
.
Hacer I ⟵ 1
Repetir con I desde 1 hasta 15
    Hacer J ⟵ 1
    Repetir con J desde 1 hasta 5
        Hacer MATRIZ [I, J] ⟵ 0 y J ⟵ J + 1
    {Fin del ciclo interno}
    Hacer I ⟵ I + 1
{Fin del ciclo externo}
.
.
.
```

Al variar los valores de I y J se asigna el 0 a cada uno de los elementos de MATRIZ.

www.FreeLibros.com

Para I = 1 y J = 1, se asigna el valor 0 al elemento del renglón 1 y columna 1.
I = 1 y J = 2, se asigna el valor 0 al elemento del renglón 1 y columna 2.
. . .

I = 15 y J = 5, se asigna el valor 0 al elemento del renglón 15 y columna 5.

En la figura 4.21 se presenta como queda el arreglo bidimensional una vez asignado el valor 0 a cada una de las casillas:

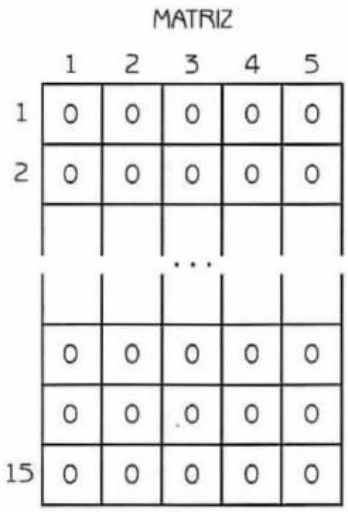


Figura 4.21 Asignación en arreglos.

2. *A un elemento en particular* del arreglo: en este caso la asignación es directa. Por ejemplo para asignar el valor 12 al elemento del renglón 13 y columna 4, hacemos:

MATRIZ[13,4] ⟵ 12

El arreglo queda como se muestra en la figura 4.22.

www.FreeLibros.com

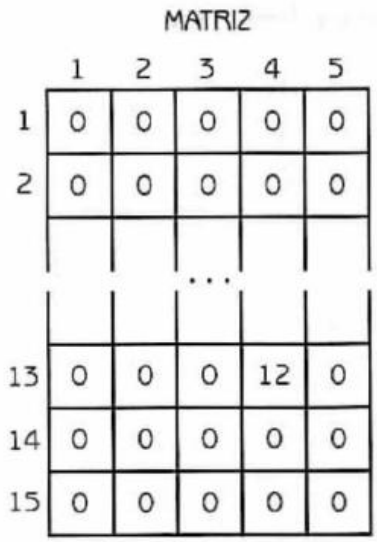


Figura 4.22 Asignación en arreglos.

Consideramos conveniente aclarar que las operaciones de lectura, escritura y asignación a todos los elementos de un arreglo bidimensional pueden hacerse por renglones (Figura 4.23),

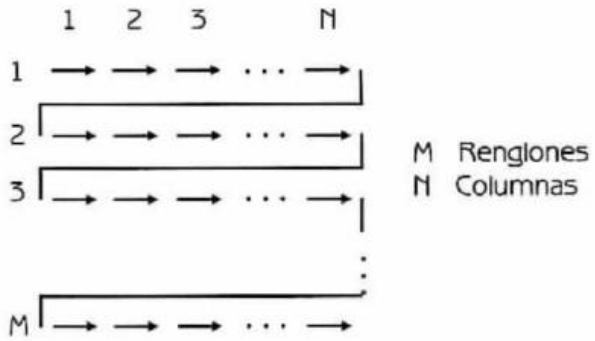


Figura 4.23 Recorrido de un arreglo bidimensional por renglones.

o por columnas (Figura 4.24).

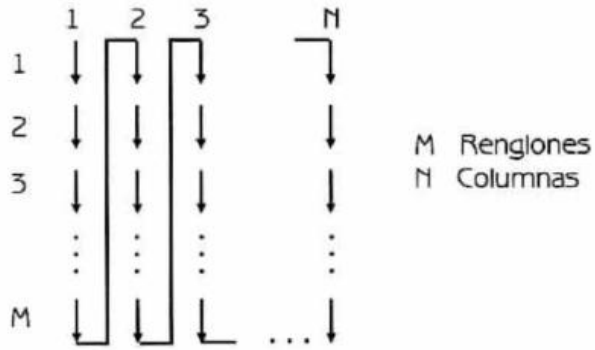


Figura 4.24 Recorrido de un arreglo bidimensional por columnas.

www.FreeLibros.com

4.3.2 Arreglos de más de dos dimensiones

Un arreglo de N dimensiones es una colección de $K_1 * K_2 * \ldots * K_N$ elementos. Para hacer referencia a cada componente de un arreglo de N dimensiones, se usarán N índices (uno para cada dimensión).

Se definirá el arreglo A de N dimensiones de la siguiente manera:

$$A = \text{ARREGLO } [LI_1..LS_1, LI_2..LS_2, \ldots, LI_N..LS_N] \text{ DE tipo}$$

Donde:

LI_1: Límite Inferior del índice 1
LS_1: Límite Superior del índice 1
. . .
LI_N: Límite Inferior del índice N
LS_N: Límite Superior del índice N

El total de elementos de A será :

$$NTE = (LS_1 - LI_1 + 1) * (LS_2 - LI_2 + 1) * \ldots * (LS_N - LI_N + 1)$$

Fórmula 4.3

Por ejemplo, el arreglo tridimensional TRI [1..3, 1..2, 1..3] tendrá:

$$(3 - 1 + 1) * (2 - 1 + 1) * (3 - 1 + 1) = 3 * 2 * 3 = 18 \text{ elementos}$$

Gráficamente puede representarse el arreglo TRI como se muestra en las figuras 4.25 y 4.26.

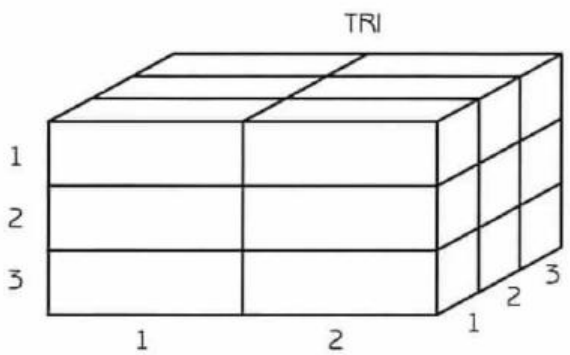


Figura 4.25 Representación de arreglos de más de dos dimensiones.

www.FreeLibros.com

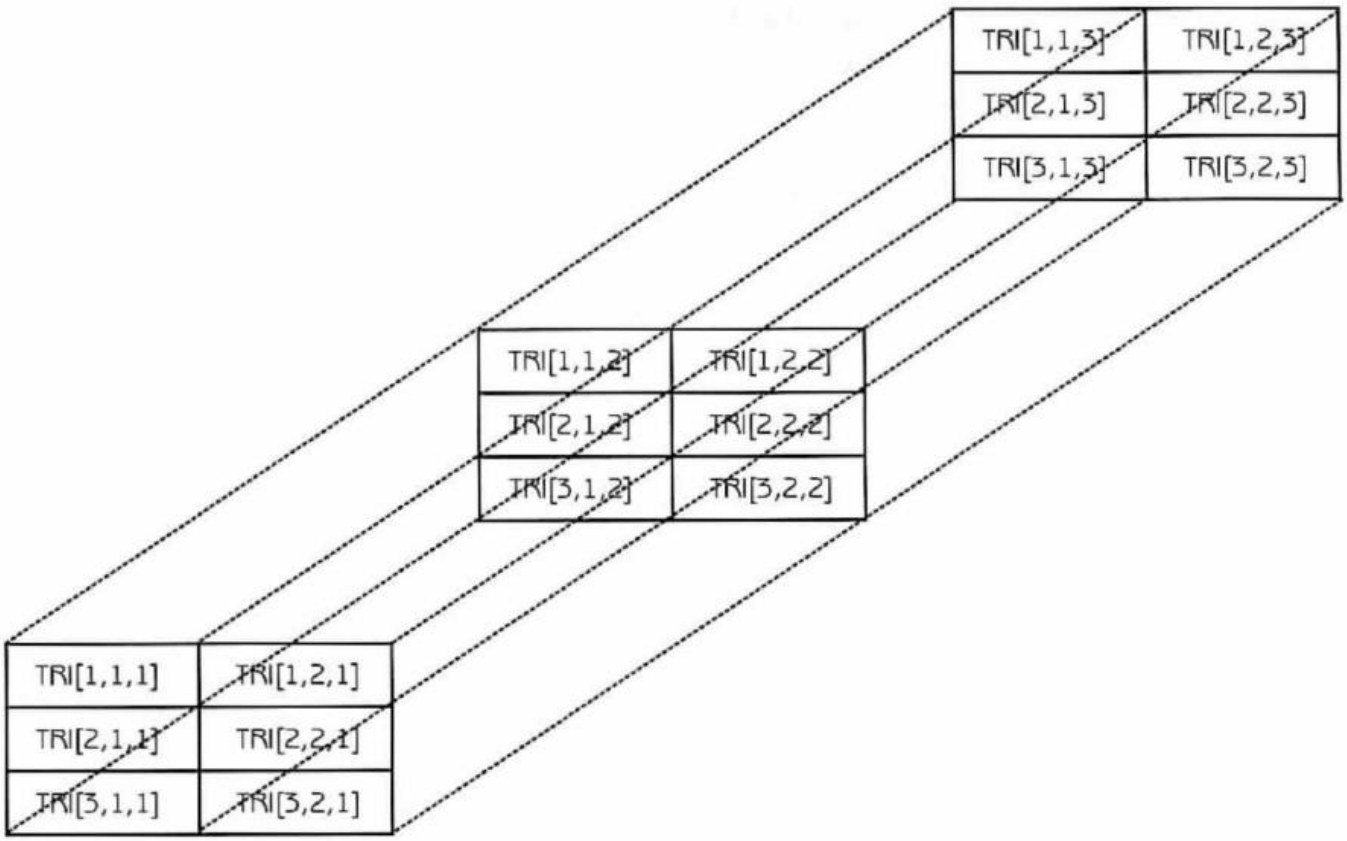


Figura 4.26 Representación de arreglos de más de dos dimensiones.

Ejemplo 4.10

El Centro Meteorológico de México lleva el control de las lluvias caídas mensualmente en los 32 estados de la república. La información se ha registrado a lo largo de los últimos 5 años. Para almacenar la información se definirá un arreglo tridimensional ($32 \times 12 \times 5 = 1920$ elementos).

MET = ARREGLO [1..32, 1..12, 1..5] DE reales

- NTE = (32 –1 + 1) * (12 –1 + 1) * (5 –1 + 1) = 1920
- Cada elemento de MET será un número real. Para hacer referencia a cada uno de ellos se utilizarán tres índices y el nombre de la variable tipo arreglo: MET [i, j, k].

Donde: $1 \leq i \leq 32$
$1 \leq j \leq 12$
$1 \leq k \leq 5$

MET [1, 5, 3] hace referencia a la lluvia caída en el estado 1, en el mes 5 del tercer año.

www.FreeLibros.com

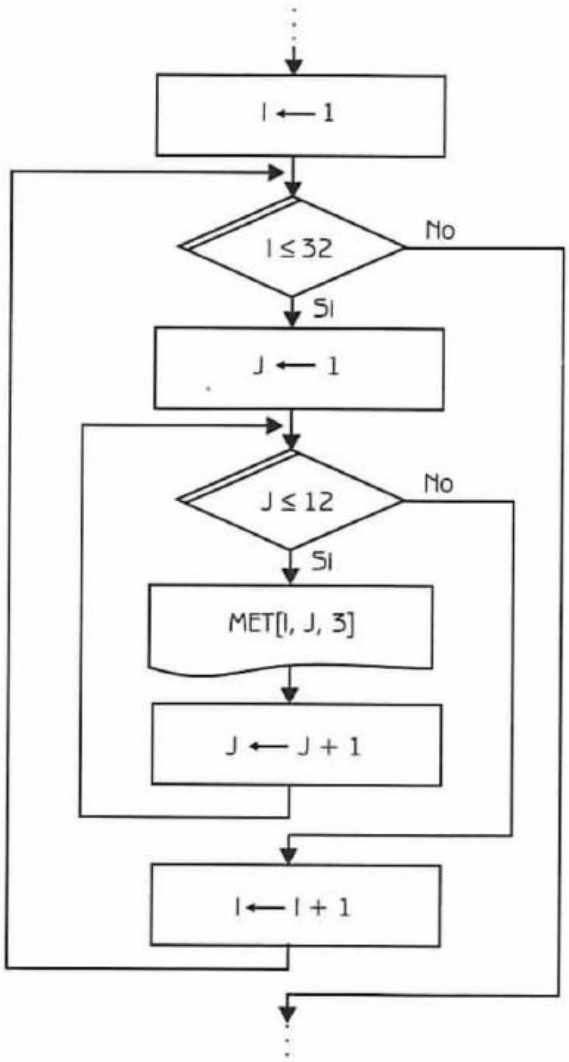
MET [21, 9, 4] hace referencia a la lluvia caída en el estado 21, en el mes 9 del cuarto año.

. . .

MET [32, 12, 5] hace referencia a la lluvia caída en el estado 32, en el mes 12 del quinto año.

Supongamos ahora que en el Centro Meteorológico necesitan obtener la siguiente información:

a) Las lluvias mensuales caídas en cada estado, durante el tercer año. Para obtener la información solicitada, realizamos las siguientes operaciones:



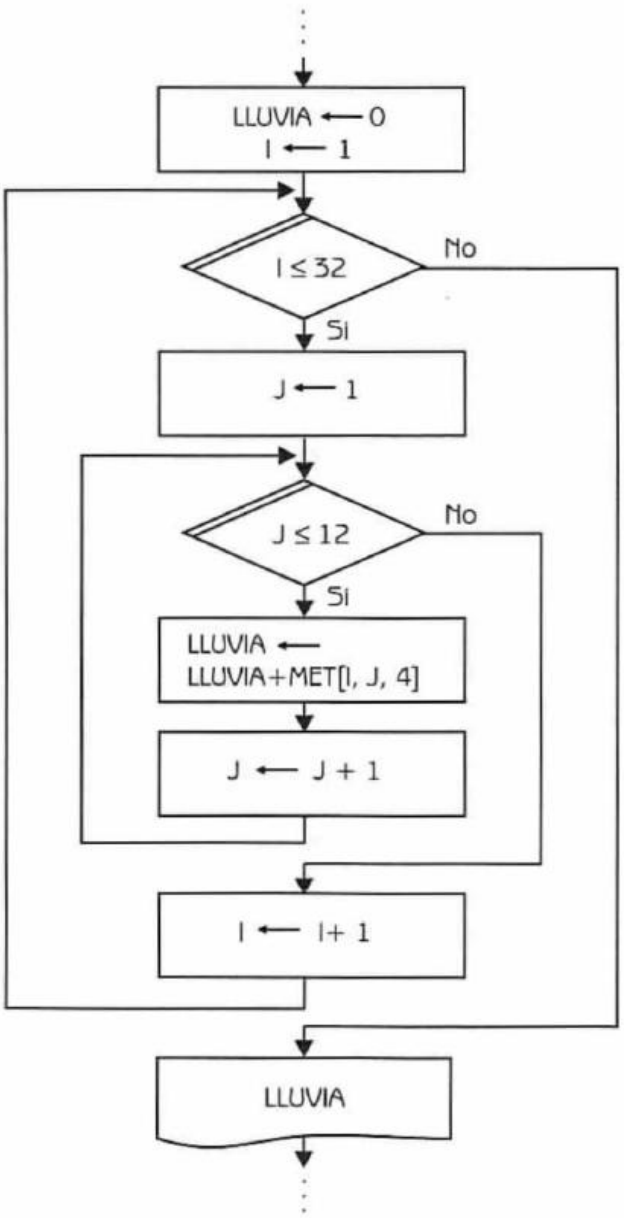

```
:
Hacer I ←—— 1
Repetir con I desde 1 hasta 32
    Hacer J ←—— 1
```

www.FreeLibros.com

```
        Repetir con J desde 1 hasta 12
            Escribir MET [I, J, 3]
            Hacer J ⟵ J + 1
        {Fin del ciclo interno}
        Hacer I ⟵ I + 1
    {Fin del ciclo externo}
    .
    .
    .
```

En este caso se asigna el número 3 al tercer índice (correspondiente a los años) y se hace variar a los otros dos índices. De esta manera se escribirán las lluvias mensuales.

b) El total de lluvias caídas en el país durante el cuarto año. Para obtener la información solicitada se deben realizar los siguientes pasos:

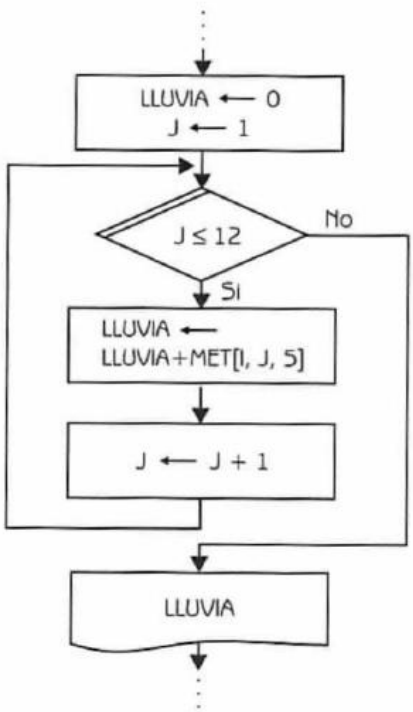


www.FreeLibros.com

```
.
.
.
Hacer LLUVIA ←—— 0 e I ←—— 1
Repetir con I desde 1 hasta 32
    Hacer J ←—— 1
    Repetir con J desde 1 hasta 12
      Hacer LLUVIA ←—— LLUVIA + MET [I, J, 4] y J ←—— J + 1
    {Fin del ciclo interno}
    Hacer I ←—— I + 1
{Fin del ciclo externo}
Escribir LLUVIA
.
.
.
```

Este caso es similar al anterior. La diferencia radica en que las lluvias mensuales de cada estado no se escribirán, sino que se acumularán obteniendo un total anual.

c) El total de lluvias caídas durante el último año del estado 17. Para obtener la información solicitada será necesario ejecutar los siguientes pasos:

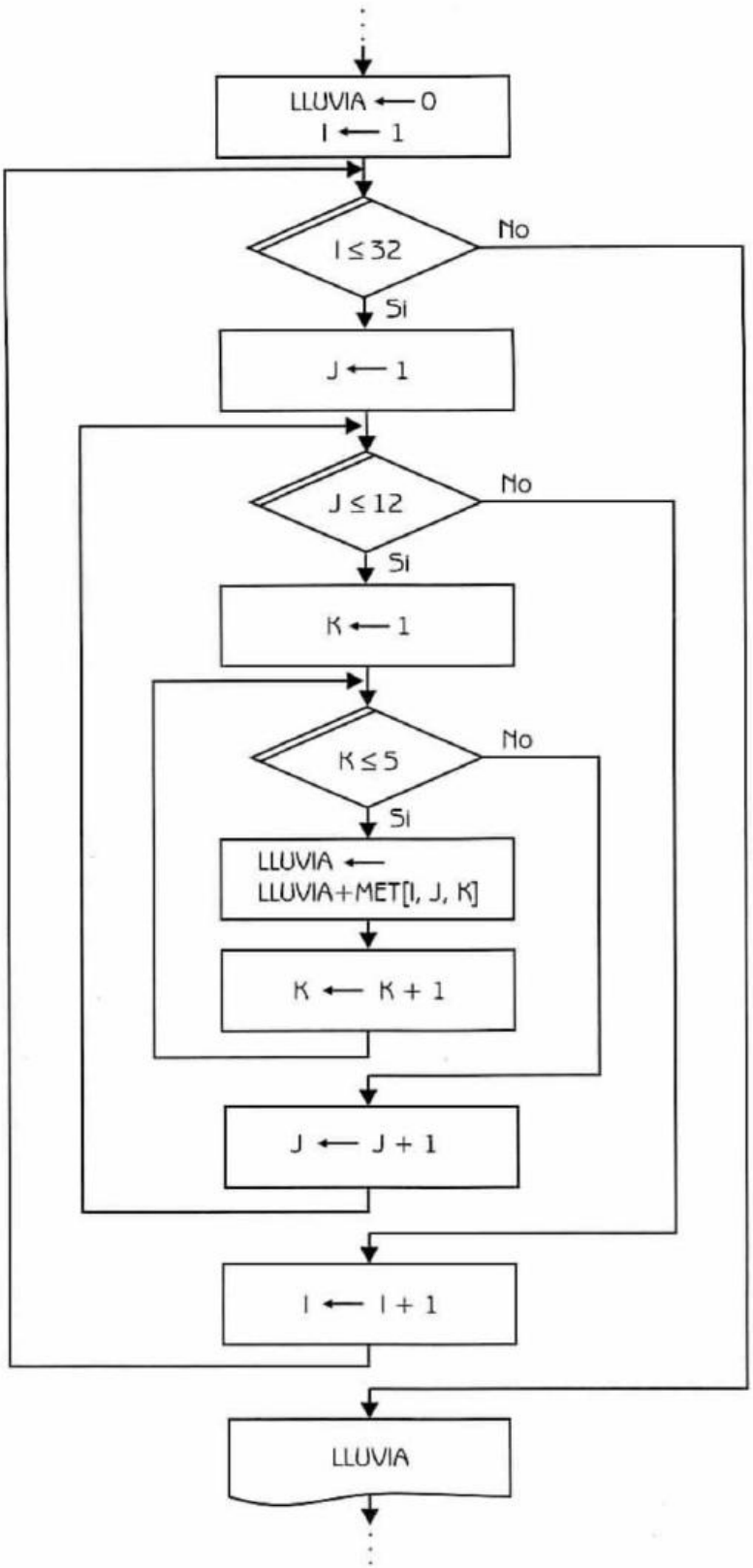



```
.
.
.
Hacer LLUVIA ←—— 0 y J ←—— 1
Repetir con J desde 1 hasta 12
    Hacer LLUVIA ←—— LLUVIA + MET[17, J, 5] y J ←—— J + 1
{Fin del ciclo}
Escribir LLUVIA
.
.
.
```

www.FreeLibros.com

En este caso se tienen dos índices constantes, el del estado (17) y el de años (5) y se hace variar solamente el índice de meses. Concluido el ciclo se escribirá el total anual de lluvias en el estado 17, en el último año.

d) El total de lluvias en México en los últimos 5 años.



www.FreeLibros.com

```
.
.
.
Hacer LLUVIA ←—— 0 e I ←—— 1
Repetir con I desde 1 hasta 32
    Hacer J ←—— 1
    Repetir con J desde 1 hasta 12
        Hacer K ←—— 1
        Repetir con K desde 1 hasta 5
            Hacer LLUVIA ←—— LLUVIA + MET [I, J, K] y K ←—— K + 1
        {Fin del ciclo más interno}
        Hacer J ←—— J + 1
    {Fin del ciclo interno}
    Hacer I ←—— I + 1
{Fin del ciclo externo}
Escribir LLUVIA
.
.
.
```

En este caso se hacen variar los tres índices, cuando concluyen los ciclos se escribe el total de lluvias caídas en el país, en los últimos 5 años.

www.FreeLibros.com

Problemas resueltos

Problema 4.1

Escriba un diagrama de flujo que reciba como entrada un arreglo unidimensional ordenado de enteros (posiblemente repetidos) y genere como salida una lista de los números enteros, pero sin repeticiones.

Dato: VEC [1..N] $1 \leq N \leq 500$

Donde:

VEC es un arreglo unidimensional de enteros, cuya capacidad máxima es de 500 elementos.

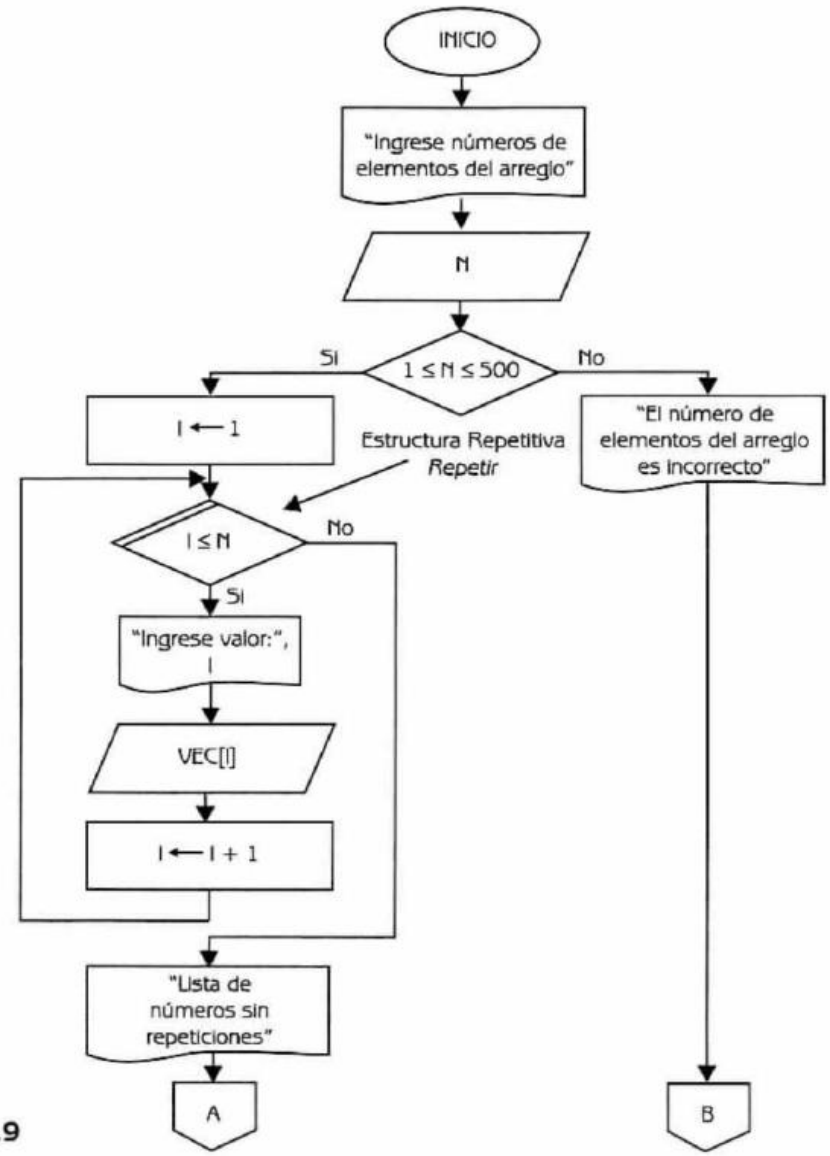


Diagrama de Flujo 4.9

(continúa)

www.FreeLibros.com

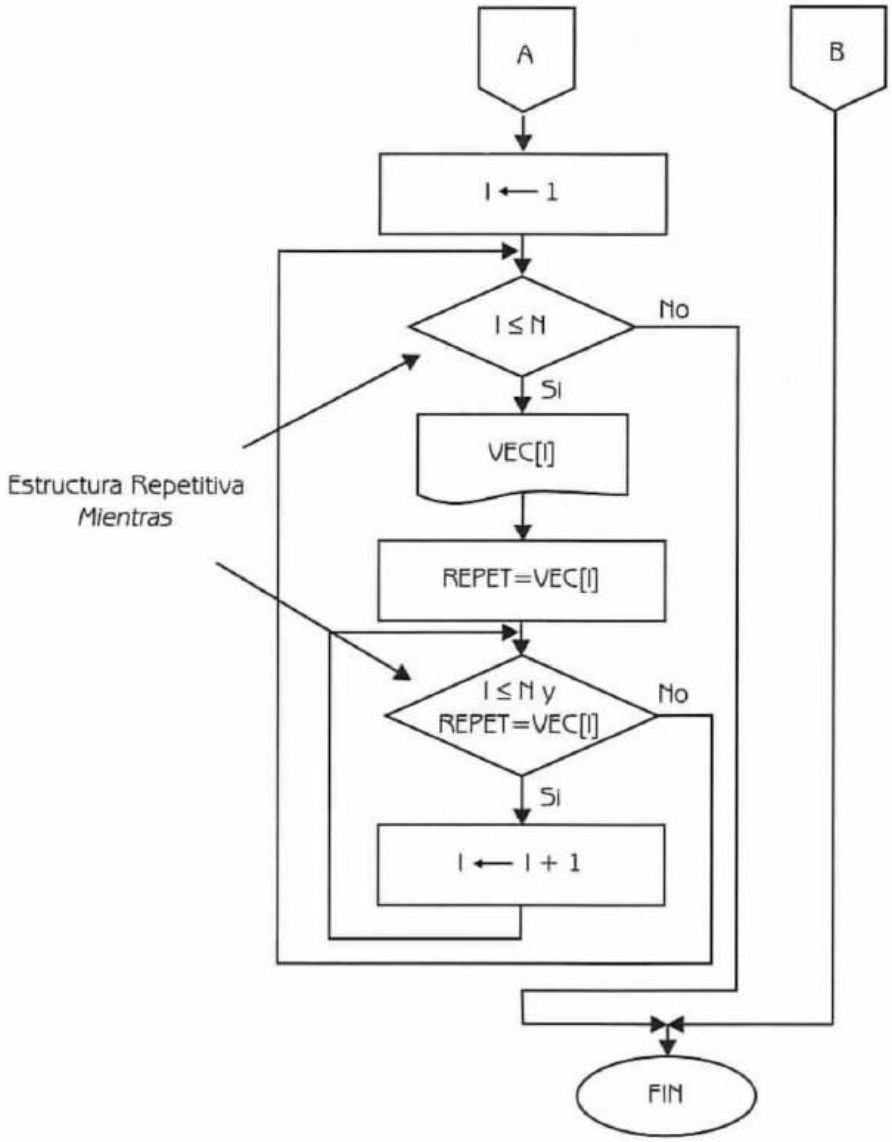


Diagrama de Flujo 4.9 (*continuación*)

Explicación de variables

N: Variable de tipo entero. Indica el número de elementos que se recibirán como datos. Su valor debe estar comprendido en el intervalo [1..500].

I: Variable de tipo entero. Se usa como índice del arreglo y como variable de control en diferentes ciclos.

VEC: Arreglo unidimensional de tipo entero.

REPET: Variable de tipo entero. Se utiliza como auxiliar para almacenar el valor impreso. Con este valor se comparan los siguientes números para determinar si hay repetidos.

A continuación presentamos el programa correspondiente.

www.FreeLibros.com

Programa 4.9

```
LISTA_SIN_REPETICIONES

{El programa, dado como entrada un vector ordenado de enteros, proporciona
como salida una lista de los números ordenados pero sin repeticiones}

{N, I y REPET son variables de tipo entero. VEC es un arreglo unidimensional de tipo
entero}

1. Escribir "Ingresa número de elementos del arreglo:"
2. Leer N
3. Si (N >= 1) y (N <= 500)
        entonces
                Hacer I ←—— 1
          3.1 Repetir con I desde 1 hasta N
                        Escribir "Ingrese valor: ", I
                        Leer VEC[I]
                        Hacer I ←—— I+1
           3.2 {Fin del ciclo del paso 3.1}
               Escribir "Lista de números sin repeticiones"
               Hacer I ←—— 1
           3.3 Mientras (I <= N) repetir
                        Escribir VEC[I]
                        Hacer REPET ←—— VEC[I]
                 3.3.1 Mientras (I <= N) y (REPET=VEC[I]) repetir
                                Hacer I ←—— I+1
                 3.3.2 {Fin del ciclo del paso 3.3.1}
           3.4 {Fin del ciclo del paso 3.3}
        sino
                Escribir "El número de elementos del arreglo es incorrecto"
4. {Fin del condicional del paso 3}
```

Problema 4.2

Escriba un diagrama de flujo que reciba como entrada un arreglo unidimensional desordenado de enteros (posiblemente repetidos) y obtenga como salida ese mismo vector, pero sin repeticiones.

Dato: VEC[1..N] $1 \leq N \leq 500$

www.FreeLibros.com

Donde: VEC es un arreglo unidimensional de enteros que tiene como máximo 500 elementos.

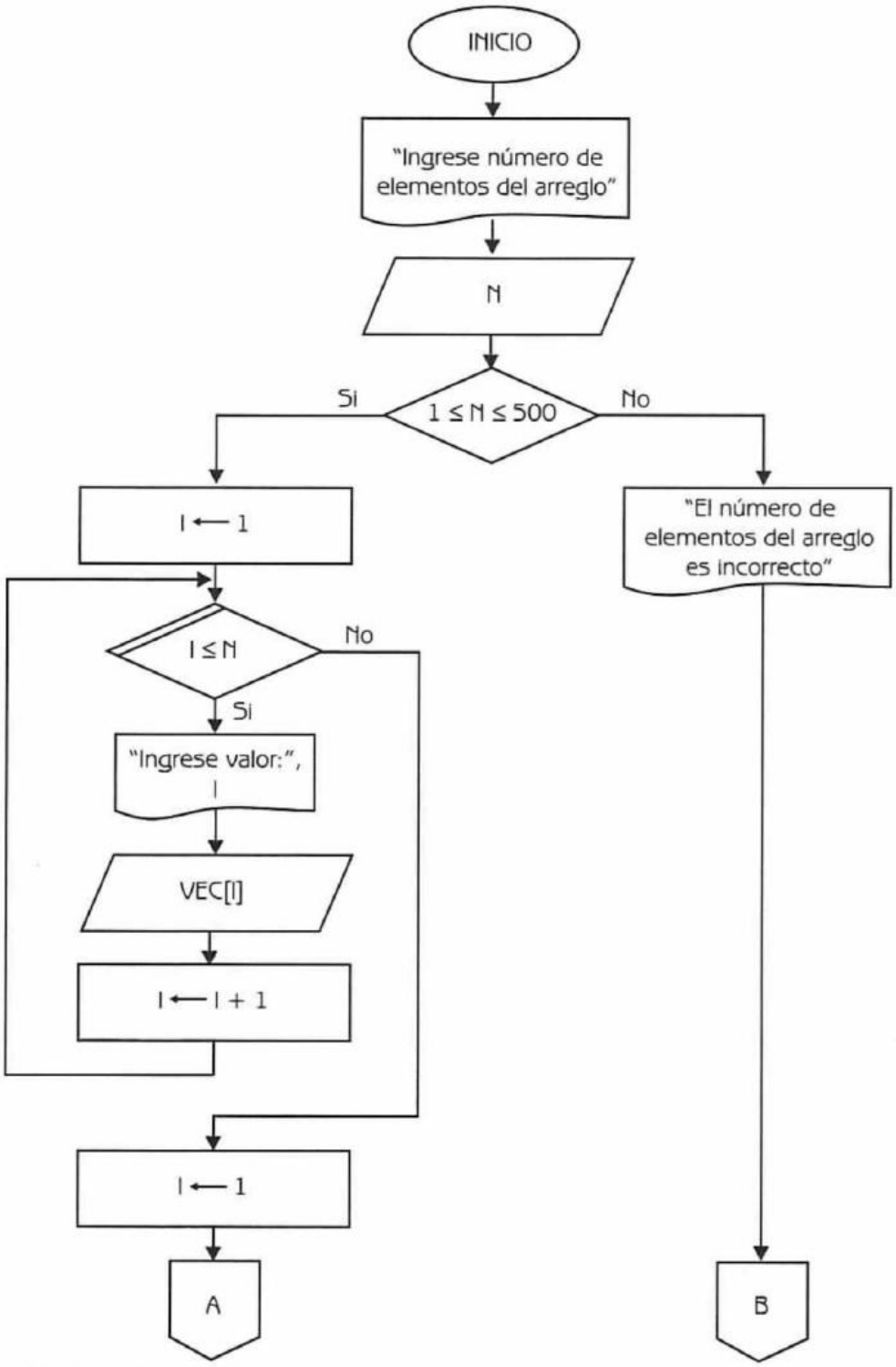


Diagrama de Flujo 4.10 (*continúa*)

www.FreeLibros.com

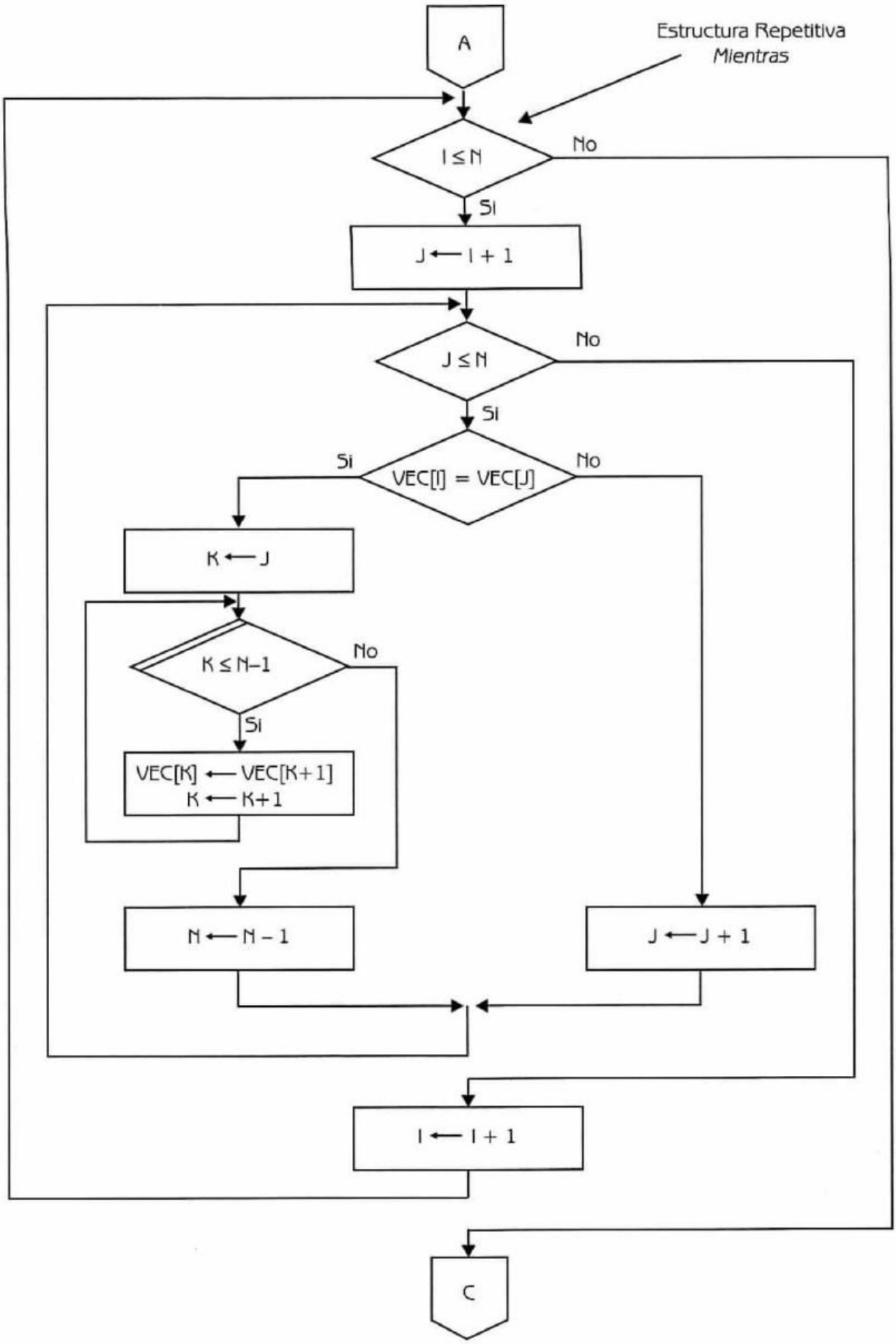


Diagrama de Flujo 4.10 (*continuación...*)

www.FreeLibros.com

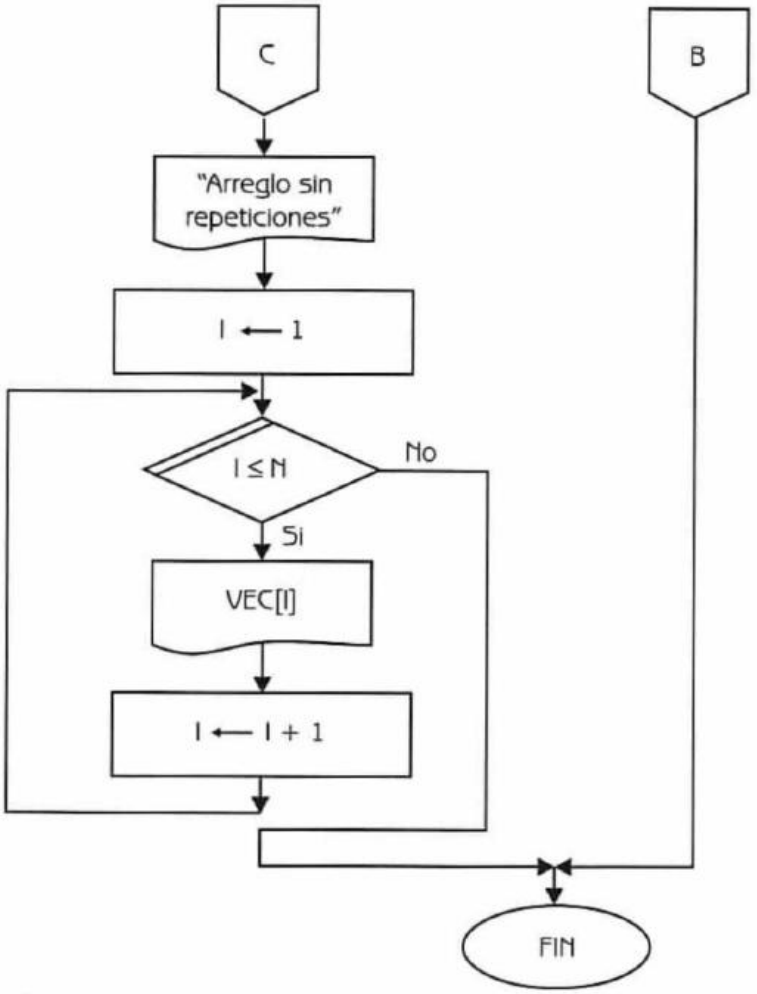


Diagrama de Flujo 4.10 (*continuación*)

Explicación de variables

N: Variable de tipo entero. Indica el total de valores que se recibirán como datos. Debe ser un número menor o igual a 500 y mayor o igual a 1.

I: Variable de tipo entero. Se utiliza como índice del arreglo y como variable de control de diferentes ciclos.

VEC: Arreglo unidimensional de tipo entero.

K: Variable de tipo entero. Se usa como variable de control del ciclo en el cual se hace la eliminación del elemento repetido.

A continuación presentamos el diagrama de flujo en lenguaje algorítmico.

www.FreeLibros.com

Programa 4.10

```
VECTOR_SIN_REPETICIONES{El programa, dado como entrada un vector ordenado
de enteros, obtiene como salida el mismo vector pero sin repeticiones}

{N, I y K son variables de tipo entero. VEC es un arreglo unidimensional de tipo entero}

1. Escribir "Ingrese el número de elementos del arreglo:"
2. Leer N
3. Si (N >= 1) y (N <= 500)
        entonces
                Hacer I ⟵ 1
          3.1 Repetir con I desde 1 hasta N
                        Escribir "Ingrese valor:", I
                        Leer VEC[I]
                        Hacer I ⟵ I+1
          3.2 {Fin del ciclo del paso 3.1}
                Hacer I ⟵ 1
          3.3 Mientras I <= N repetir
                        Hacer J ⟵ I+1
                3.3.1 Mientras J <= N repetir
                                3.3.1.1 Si VEC[I]=VEC[J]
                                        entonces
                                                Hacer K ⟵ J
                                        3.3.1.1.1 Repetir con K desde J hasta N-1
                                                Hacer VEC[K] ⟵ VEC[K+1] y
                                                        K ⟵ K+1
                                        3.3.1.1.2 {Fin del ciclo del paso 3.3.1.1.1}
                                                Hacer N ⟵ N-1
                                           sino
                                                Hacer J ⟵ J+1
                                3.3.1.2 {Fin del condicional del paso 3.3.1.1}
                3.3.2 {Fin del ciclo del paso 3.3.1}
                        Hacer I ⟵ I+1
          3.4 {Fin del ciclo del paso 3.3}
                Escribir "Arreglo sin repeticiones"
                Hacer I ⟵ 1
          3.5 Repetir con I desde 1 hasta N
                        Escribir VEC[I]
                        Hacer I ⟵ I+1
          3.6 {Fin del ciclo del paso 3.5}
        sino
                Escribir "El número de elementos del arreglo es incorrecto"
4. {Fin del condicional del paso 3}
```

www.FreeLibros.com

Problema 4.3

Supongamos que en una elección hubo 12 candidatos (con identificadores 1,2,3,...,12). Por otra parte, los votos para cada candidato se teclean de manera desorganizada como se muestra a continuación:

1 5 7 5 1 12 10 7 1 7 5 8 1 5 -1

Nota: El final de los datos está dado por –1.

Construya un diagrama de flujo que pueda proporcionar la siguiente información:

a) El número de votos de cada candidato al final de la elección.

b) El candidato ganador, el número de votos que obtuvo y el porcentaje correspondiente del total de la elección. Suponemos que el candidato ganador no empató en número de votos con otro candidato.

Datos: $VOTO_1$, $VOTO_{2,}$,...,–1

Donde:

$VOTO_i$ representa el número de candidato por el que votó la persona i. Es una variable de tipo entero.

www.FreeLibros.com

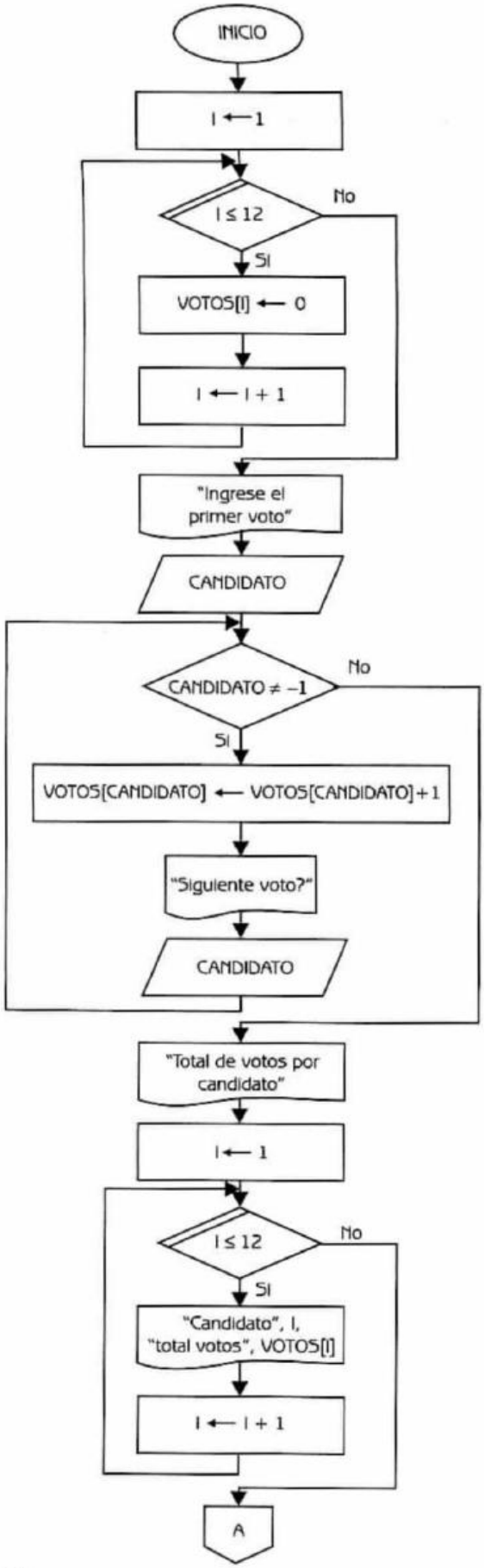


Diagrama de Flujo 4.11 (*continúa*)

www.FreeLibros.com

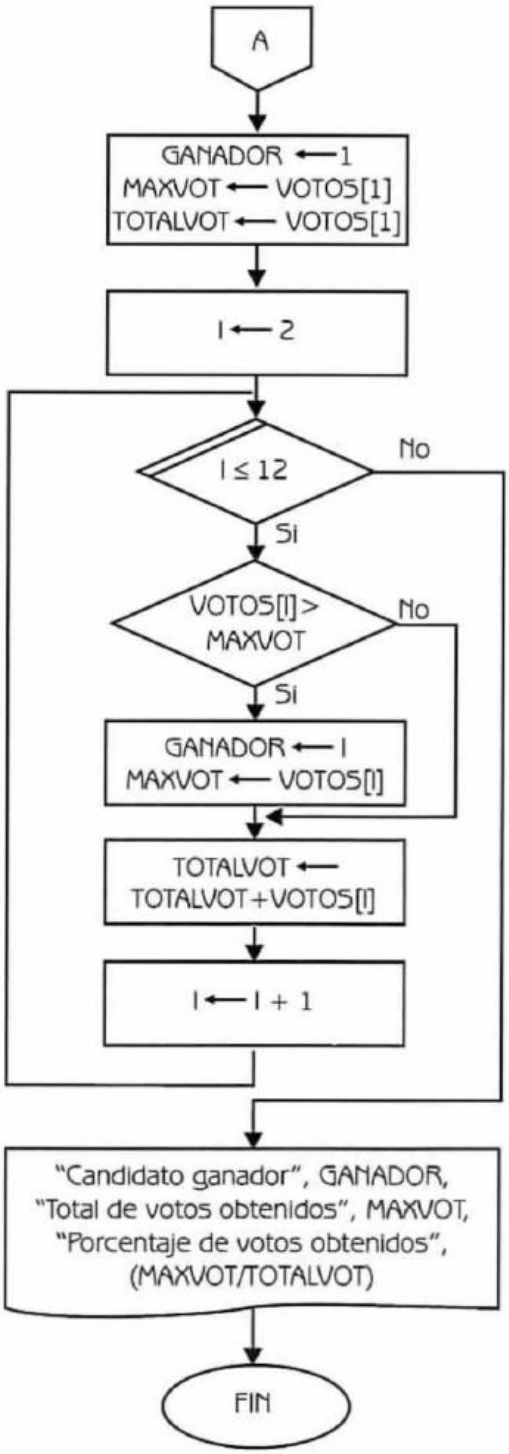


Diagrama de Flujo 4.11 (*continuación*)

Explicación de las variables

I: Variable de tipo entero. Se utiliza como variable de control en diferentes ciclos.

VOTOS: Arreglo unidimensional de tipo entero. Almacena el total de votos obtenidos por cada uno de los 12 candidatos.

www.FreeLibros.com

CANDIDATO: Variable de tipo entero. Se utiliza para ingresar el número del candidato votado.

GANADOR: Variable de tipo entero. Se utiliza para almacenar el número de candidato que más votos obtuvo.

MAXVOT: Variable de tipo entero. Se utiliza para almacenar el total de votos obtenidos por el candidato ganador.

TOTALVOT: Variable de tipo entero. Sirve para acumular los votos y calcular posteriormente el porcentaje de votos obtenidos por el candidato ganador.

A continuación presentamos el diagrama de flujo en lenguaje algorítmico.

Programa 4.11

```
ELECCION

{El programa, dados los votos para los diferentes candidatos de una elección, obtiene el número de votos de cada candidato al final de la elección, el candidato ganador, el número de votos que obtuvo y el porcentaje correspondiente del total de la elección}

{I, CANDIDATO, GANADOR, MAXVOT, y TOTALVOTOS son variables de tipo entero. VOTOS es un arreglo unidimensional de tipo entero}

 1. Hacer I ←—— 1
 2. Repetir con I desde 1 hasta 12
        Hacer VOTOS[I] ←—— 0 e I ←—— I+1
 3. {Fin del ciclo del paso 2}
 4. Escribir "Ingrese el primer voto:"
 5. Leer CANDIDATO
 6. Mientras (CANDIDATO <> –1) repetir
        Hacer VOTOS[CANDIDATO] ←—— VOTOS[CANDIDATO] + 1
        Escribir "Siguiente voto?"
        Leer CANDIDATO
 7. {Fin del ciclo del paso 6}
 8. Escribir "Total de votos por candidato"
 9. Hacer I ←—— 1
10. Repetir con I desde 1 hasta 12
                Escribir "Candidato", I, "total de votos", VOTOS[I]
                Hacer I ←—— I+1
```

www.FreeLibros.com

```
11. {Fin del ciclo del paso 10}
12. Hacer GANADOR ⟵ 1, MAXVOT ⟵ VOTOS[1],
        TOTALVOT ⟵ VOTOS[1] e I ⟵ 2
13. Repetir con I desde 2 hasta 12
        13.1 Si VOTOS[I] > MAXVOT entonces
                Hacer GANADOR ⟵ I, MAXVOT ⟵ VOTOS[I]
        13.2 {Fin del condicional del paso 13.1}
             Hacer TOTALVOT ⟵ TOTALVOT + VOTOS[I] e I ⟵ I+1
14. {Fin del ciclo del paso 13}
15. Escribir "Candidato ganador", GANADOR,
             "Total de votos obtenidos", MAXVOT,
             "Porcentaje de votos obtenidos", (MAXVOT/TOTALVOT)
```

Problema 4.4

Construya un diagrama de flujo para almacenar en un arreglo unidimensional los primeros 30 números primos. Al final imprima el arreglo correspondiente.

www.FreeLibros.com

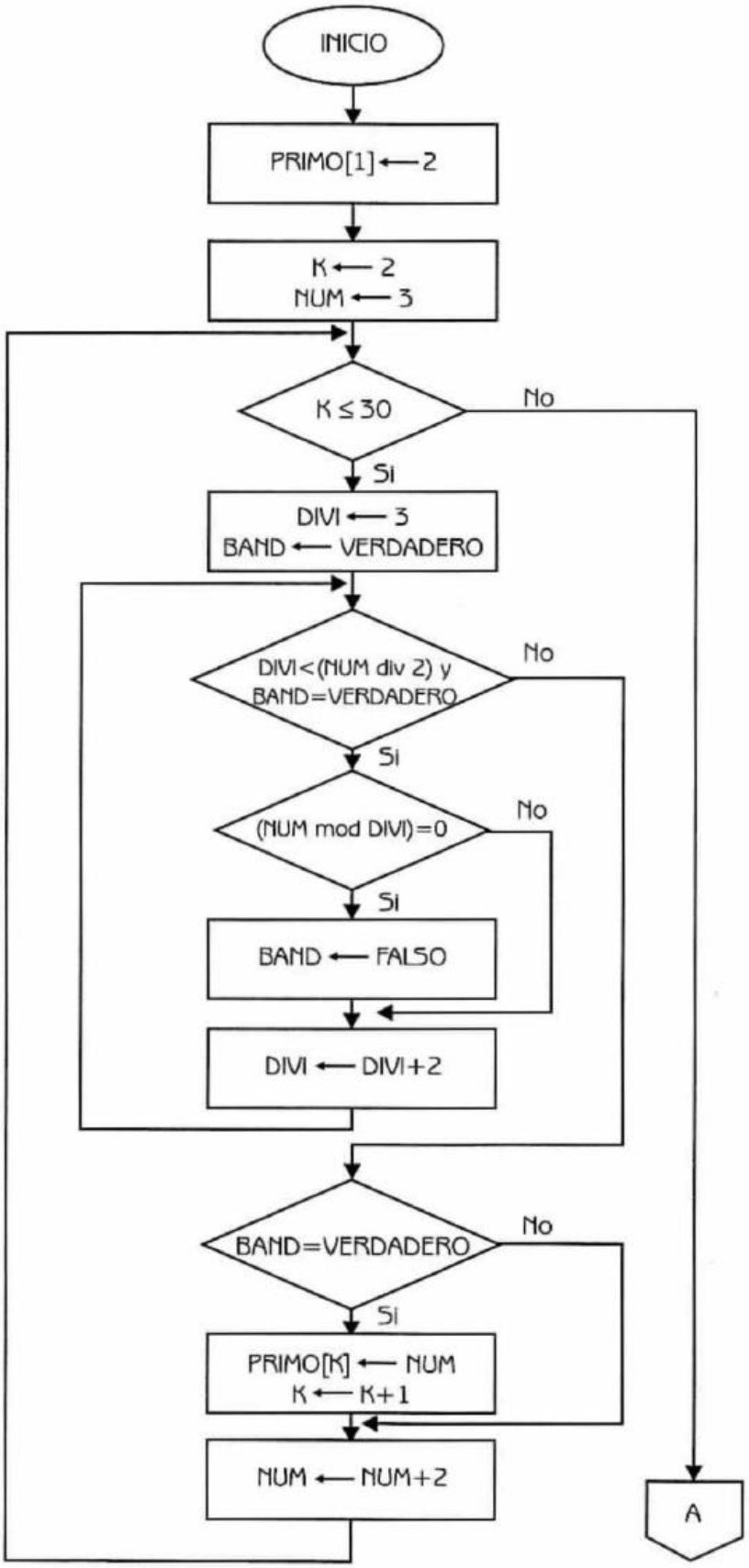


Diagrama de Flujo 4.12 (*continúa*)

www.FreeLibros.com

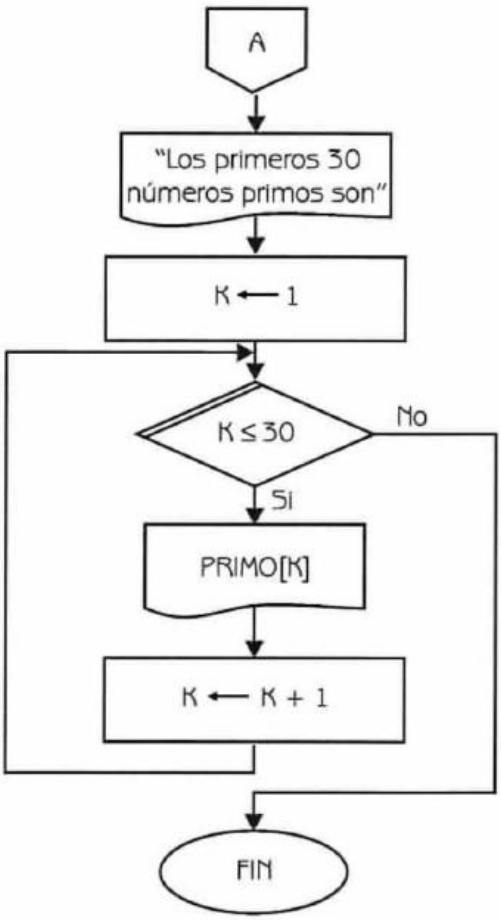


Diagrama de Flujo 4.12 (*continuación*)

Explicación de las variables

PRIMO: Arreglo unidimensional de tipo entero. Se utiliza para almacenar los primeros 30 números primos.

K: Variable de tipo entero. Se usa como índice del arreglo y como variable de control de un ciclo.

NUM: Variable de tipo entero. Almacena el número sobre el que hay que determinar si es o no un número primo.

DIVI: Variable de tipo entero. Representa los posibles divisores del número que se está analizando.

A continuación presentamos el programa correspondiente.

www.FreeLibros.com

Programa 4.12

```
ARREGLO_DE_PRIMOS

{El programa almacena en un arreglo unidimensional los primeros 30 números
primos}

{K, NUM y DIVI son variables de tipo entero. PRIMO es un arreglo unidimensional
de tipo entero. BAND es una variable de tipo booleano}

1. Hacer PRIMO[1] ←— 2, K ←— 2 y NUM ←— 3
2. Mientras (K <= 30) repetir
            Hacer DIVI ←— 3 y BAND ←— VERDADERO
      2.1 Mientras (DIVI < (NUM div 2) y BAND=VERDADERO) repetir
          2.1.1 Si ((NUM mod DIVI)=0) entonces
                      Hacer BAND ←— FALSO
          2.1.2 {Fin del condicional del paso 2.1.1}
                      Hacer DIVI ←— DIVI+2
      2.2 {Fin del ciclo del paso 2.1}
      2.3 Si (BAND=VERDADERO) entonces
          Hacer PRIMO[K] ←— NUM y K ←— K + 1
      2.4 {Fin del condicional del paso 2.3}
          Hacer NUM ←— NUM+2
3. {Fin del ciclo del paso 2}
4. Escribir "Los primeros 30 números primos son:"
5. Hacer K ←— 1
6. Repetir con K desde 1 hasta 30
            Escribir PRIMO[K]
            Hacer K ←— K+1
7. {Fin del ciclo del paso 6}
```

Problema 4.5

Ordenación por inserción directa

El método de ordenación por inserción directa es el que generalmente utilizan los jugadores de carta cuando ordenan éstas, de ahí que también se conozca con el nombre del método de la baraja. La idea central de este algoritmo consiste en insertar un elemento del arreglo en la parte izquierda del mismo, que ya se encuen-

www.FreeLibros.com

tra ordenada. Este proceso se repite desde el segundo hasta el n-ésimo elemento. Veamos a continuación un ejemplo.

Ejemplo 4.11

Supóngase que se desean ordenar las siguientes claves del arreglo UNI, utilizando el método de inserción directa.

UNI: 30 92 23 31 59 42 27 50

Las comparaciones que se realizan son las siguientes:

Primera pasada

UNI [2] < UNI [1] (92 < 30) no hay intercambio

UNI: [30] [92] 23 31 59 42 27 50

Segunda pasada

UNI [3] < UNI [2] (23 < 92) sí hay intercambio
UNI [2] < UNI [1] (23 < 30) sí hay intercambio

UNI: [23] [30] [92] 31 59 42 27 50

Tercera pasada

UNI [4] < UNI [3] (31 < 92) sí hay intercambio
UNI [3] < UNI [2] (31 < 30) no hay intercambio

UNI: [23] [30] [31] [92] 59 42 27 50

Obsérvese que una vez que se determina la posición correcta del elemento se interrumpen las comparaciones. Por ejemplo, en el caso anterior no se realizó la comparación UNI [2] < UNI [1]. En la tabla 4.3 se presenta el resultado de las restantes pasadas:

www.FreeLibros.com

Tabla 4.3								
4ta. pasada	23	30	31	59	92	42	27	50
5ta. pasada	23	30	31	42	59	92	27	50
6ta. pasada	23	27	30	31	42	59	92	50
7ma. pasada	23	27	30	31	42	50	59	92

Construya el diagrama de flujo correspondiente.

Dato: VECTOR [1..N] $1 \leq N \leq 50$ (Arreglo unidimensional de tipo entero)

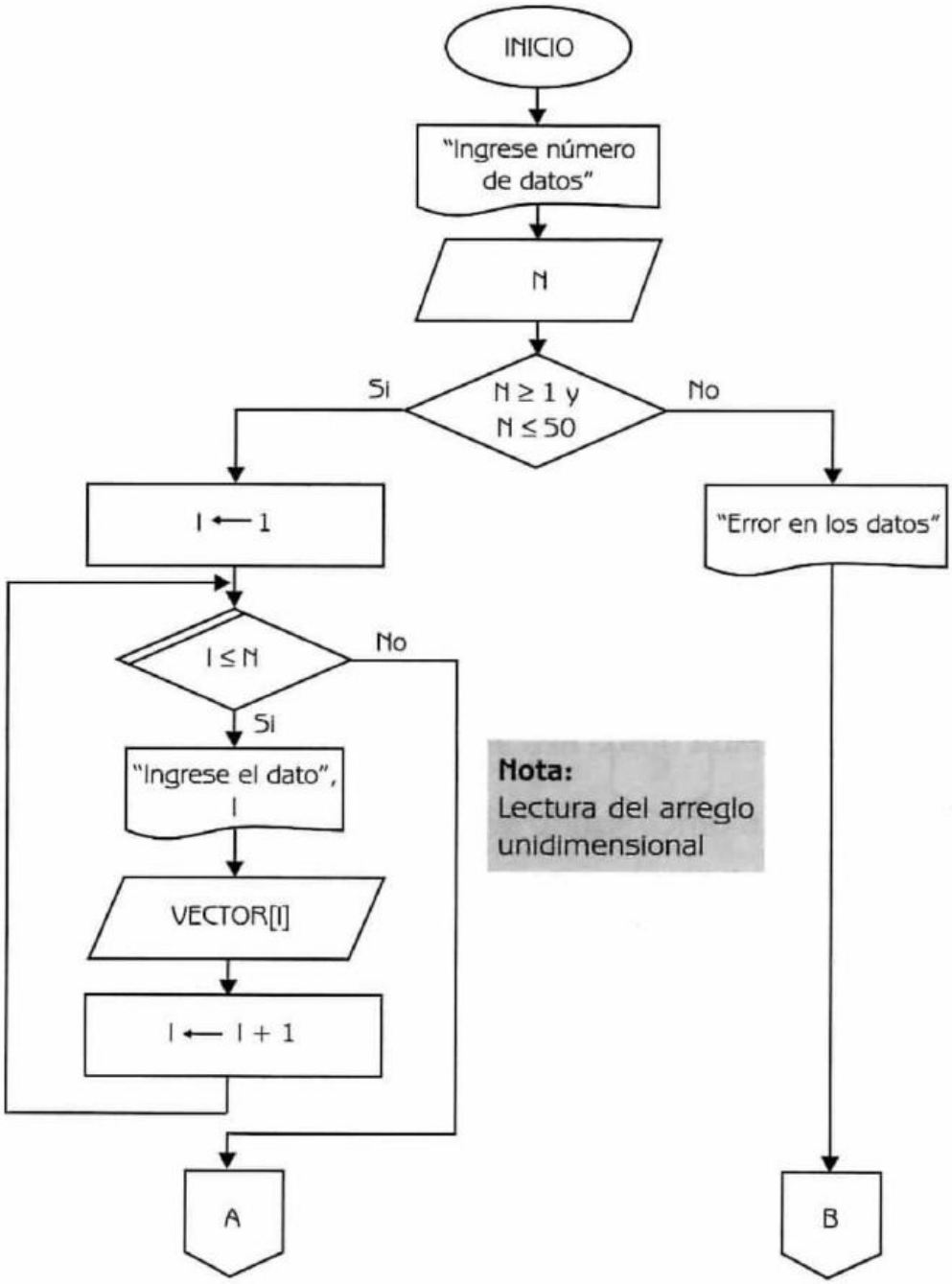
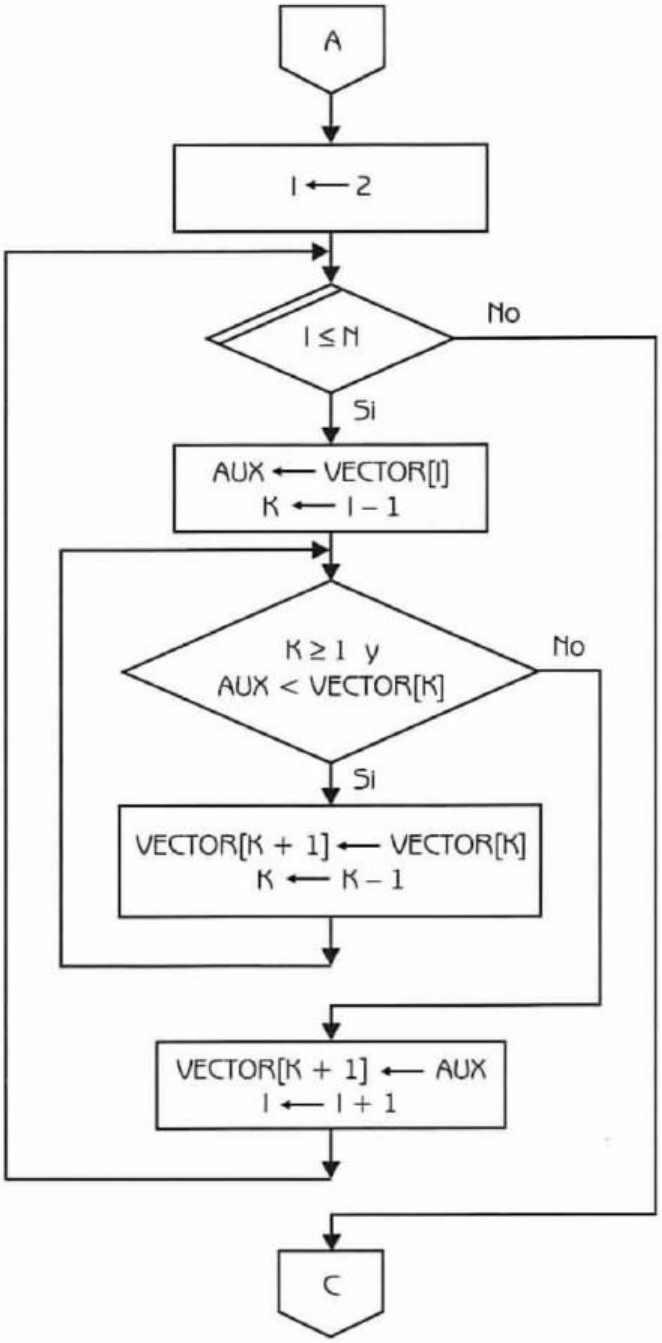


Diagrama de Flujo 4.13 (*continúa*)

www.FreeLibros.com



Nota:
Ordenación del arreglo unidimensional

Diagrama de Flujo 4.13 (*continuación...*)

Explicación de las variables

N: Variable de tipo entero. Expresa el número de elementos del arreglo a ordenar.
VECTOR: Arreglo unidimensional de tipo entero.
I: Variable de tipo entero. Se utiliza como variable de control de ciclos y como índice del arreglo.

www.FreeLibros.com

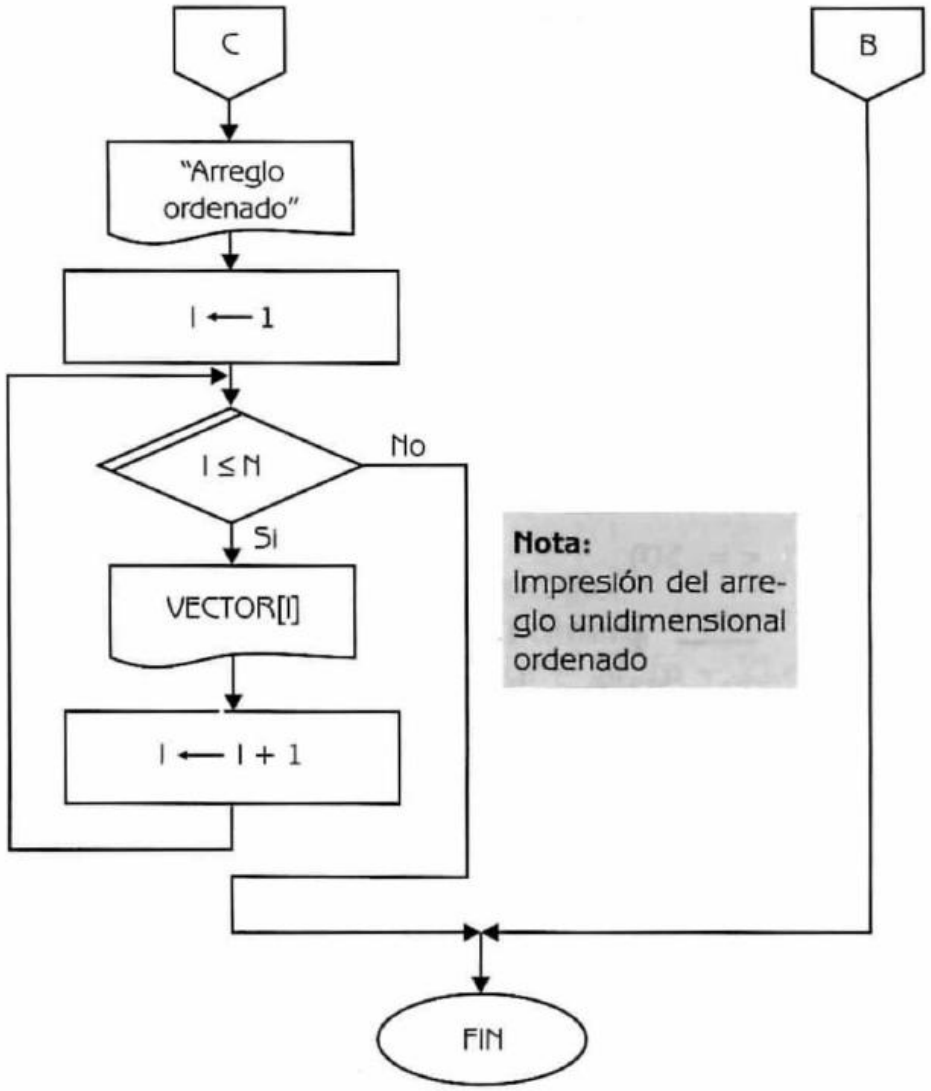


Diagrama de Flujo 4.13 (*continuación*)

AUX: Variable de tipo entero. Se usa como auxiliar en el proceso de ordenación.

K: Variable de tipo entero. Se utiliza como auxiliar en el proceso de ordenación.

A continuación presentamos el programa correspondiente.

www.FreeLibros.com

Programa 4.13

```
INSERCION_DIRECTA

{El programa ordena los elementos de un arreglo utilizando el método de inserción
directa}

{N, I, AUX, y K son variables de tipo entero. VECTOR es un arreglo unidimensional
de tipo entero}

1. Escribir "Ingrese número de datos"
2. Leer N
3. Si (N >= 1 y N <= 50)
        entonces
              Hacer I ←—— 1
        3.1 Repetir con I desde 1 hasta N
                    Escribir "Ingrese dato", I
                    Leer VECTOR[I]
                    Hacer I ←—— I + 1
        3.2 {Fin del ciclo del paso 3.1}
              Hacer I ←—— 2
        3.3 Repetir con I desde 2 hasta N
                 Hacer AUX ←—— VECTOR[I] y K ←—— I - 1
              3.3.1 Mientras (K >= 1) y (AUX < VECTOR[K] repetir
                       Hacer VECTOR[K + 1] ←—— VECTOR[K] y K ←—— K - 1
              3.3.2 {Fin del ciclo del paso 3.3.1}
                 Hacer VECTOR[K + 1] ←—— AUX e I ←—— I + 1
        3.4 {Fin del ciclo del paso 3.3}
              Escribir "Arreglo ordenado"
              Hacer I ←—— 1
        3.5 Repetir con I desde 1 hasta N
                 Escribir VECTOR[I]
                 Hacer I ←—— I + 1
        3.6 {Fin del ciclo del paso 3.5}
        sino
              Escribir "Error en los datos"
4. {Fin del condicional del paso 3}
```

www.FreeLibros.com

Problema 4.6

Ordenación por selección directa

El método de ordenación por **selección directa** es más eficiente que los métodos analizados anteriormente. Pero, aunque su comportamiento es mejor que el de aquéllos y su programación es fácil y comprensible, no es recomendable utilizarlo cuando el número de elementos del arreglo es medio o grande. La idea básica de este algoritmo consiste en buscar el menor elemento del arreglo y colocarlo en la primera posición. Luego se busca el segundo elemento más pequeño del arreglo y se le coloca en la segunda posición. El proceso continúa hasta que todos los elementos del arreglo hayan sido ordenados. El método se basa en los siguientes principios:

1. Seleccionar el menor elemento del arreglo.
2. Intercambiar dicho elemento con el primero.
3. Repetir los pasos anteriores con los (n − 1), (n − 2) elementos y así sucesivamente hasta que sólo quede el elemento mayor.

Ejemplo 4.12

Supóngase que desea ordenar las siguientes claves del arreglo UNI utilizando el método de selección directa.

UNI:	32	84	25	33	61	44	29	52

Las comparaciones que se realizan son las siguientes:

Primera pasada

Se realiza la siguiente asignación: MENOR ⟵ UNI[I] (32)

(MENOR < UNI [2] (32<84)	sí se cumple la condición
(MENOR < UNI [3] (32<25)	no se cumple la condición
	MENOR ⟵ UNI[3] (25)
(MENOR < UNI [4] (25<33)	sí se cumple la condición
(MENOR < UNI [5] (25<61)	sí se cumple la condición
(MENOR < UNI [6] (25<44)	sí se cumple la condición
(MENOR < UNI [7] (25<29)	sí se cumple la condición
(MENOR < UNI [8] (25<52)	sí se cumple la condición

Luego de la primera pasada, el arreglo queda de la siguiente forma:

www.FreeLibros.com

UNI: [25] 84 32 33 61 44 29 52

Obsérvese que el menor elemento del arreglo, UNI[3] (25), se intercambió con el primer elemento, UNI[1] (32), realizando solamente un movimiento.

Segunda pasada

Se realiza la siguiente asignación: MENOR ⟵ UNI[2] (84)

(MENOR < UNI [3] (84 < 32)	no se cumple la condición MENOR ⟵ UNI[3] (32)
(MENOR < UNI [4] (32 < 33)	sí se cumple la condición
(MENOR < UNI [5] (32 < 61)	sí se cumple la condición
(MENOR < UNI [6] (32 < 44)	sí se cumple la condición
(MENOR < UNI [7] (32 < 29)	no se cumple la condición MENOR ⟵ UNI[7] (29)
(MENOR < UNI [8] (29 < 52)	sí se cumple la condición

UNI: [25] [29] 32 33 61 44 84 52

Obsérvese que el segundo elemento más pequeño del arreglo, UNI[7] (29), se intercambió con el segundo elemento, UNI [2] (84).

En la tabla 4.4 se localiza el resultado de las restantes pasadas:

Tabla 4.4

3era. pasada	25	29	32	33	61	44	84	52
4ta. pasada	25	29	32	33	61	44	84	52
5ta. pasada	25	29	32	33	44	61	84	52
6ta. pasada	25	29	32	33	44	52	84	61
7ma. pasada	25	29	32	33	44	52	61	84

Construya el diagrama de flujo correspondiente.

Dato: VECTOR[1..N] $1 \leq N \leq 50$ (arreglo unidimensional de tipo entero).

www.FreeLibros.com

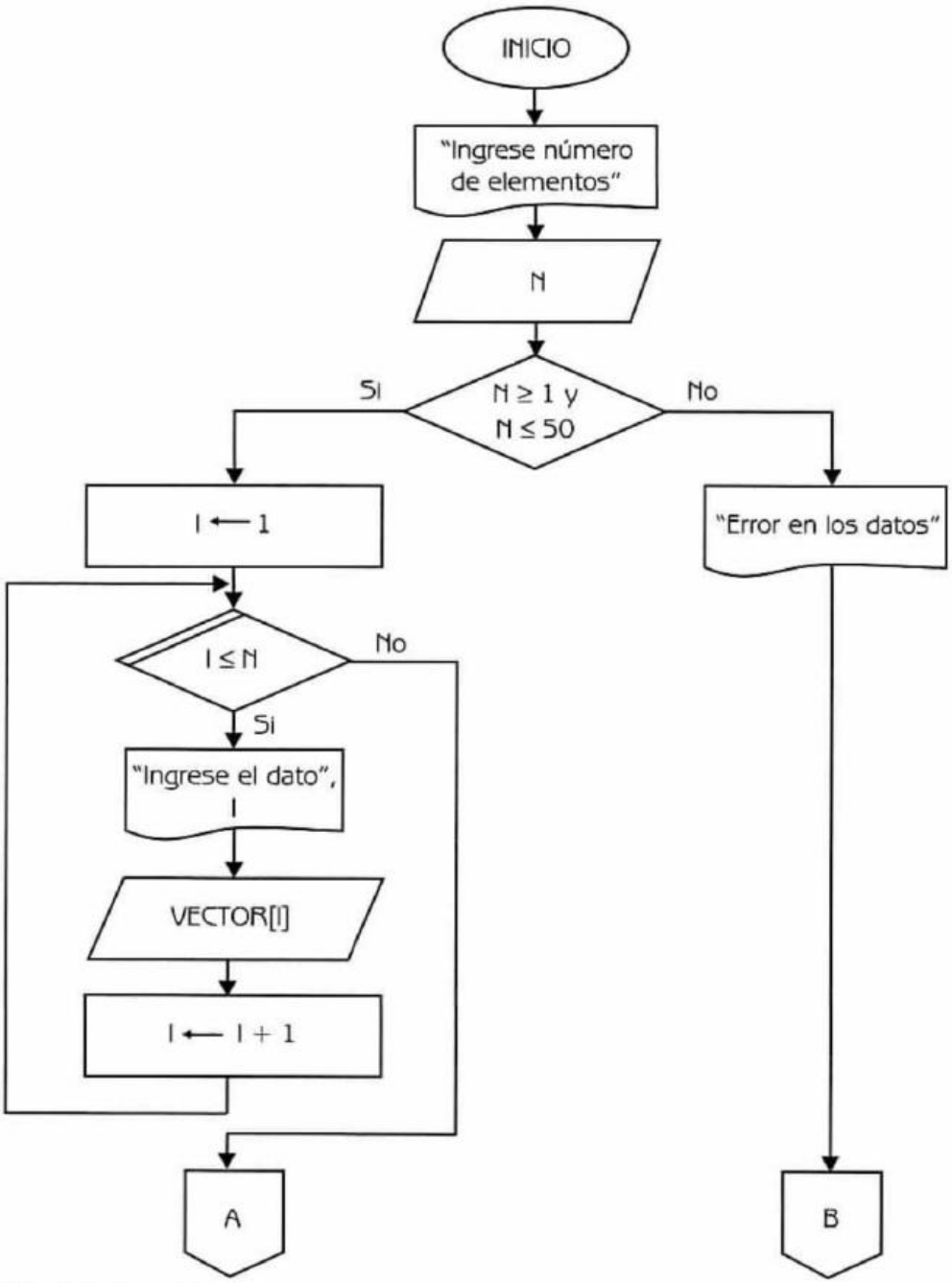


Diagrama de Flujo 4.14 (*continúa*)

www.FreeLibros.com

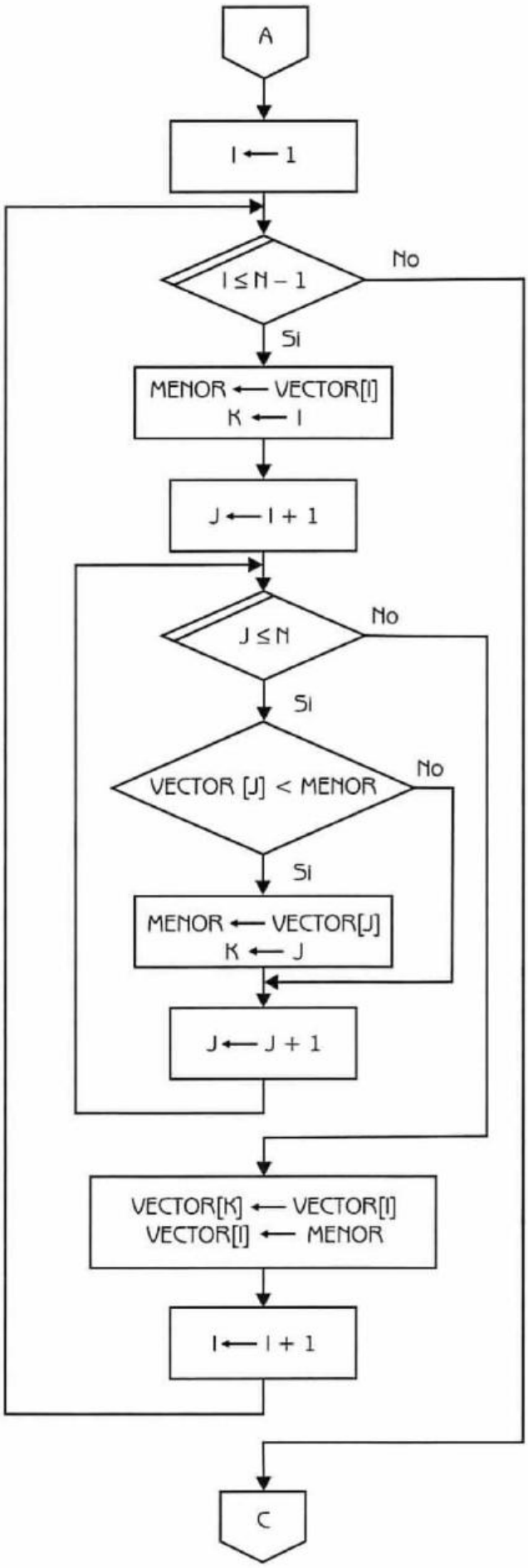


Diagrama de Flujo 4.14

(*continuación...*)

www.FreeLibros.com

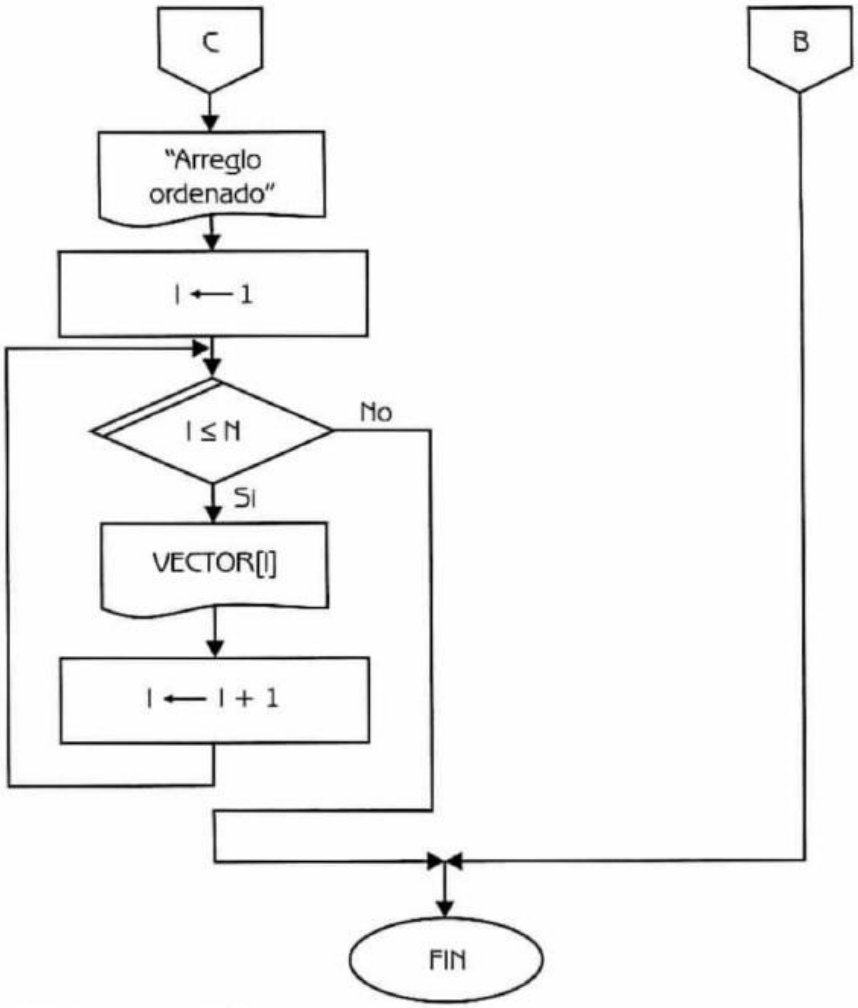


Diagrama de Flujo 4.14 (*continuación*)

Explicación de las variables

N: Variable de tipo entero. Indica el número de elementos en el arreglo a ordenar.
I: Variable de tipo entero. Se utiliza como variable de control de ciclos y como índice del arreglo.
VECTOR: Arreglo unidimensional de tipo entero.
MENOR: Variable de tipo entero. Se utiliza como auxiliar en el proceso de ordenación.
K: Variable de tipo entero. Se usa como auxiliar en el proceso de ordenación.
J: Variable de tipo entero. Se usa como auxiliar en el proceso de ordenación.

A continuación presentamos el programa correspondiente.

www.FreeLibros.com

Programa 4.14

```
SELECCION_DIRECTA

{El programa ordena un arreglo unidimensional de tipo entero por el método de
selección directa}

{N, I, MENOR, K y J son variables de tipo entero. VECTOR es un arreglo unidimen-
sional de tipo entero}

1. Escribir "Ingrese número de elementos"
2. Leer N
3. Si (N >= 1 y N <= 50)
       entonces
           Hacer I ←—— 1
     3.1 Repetir con I desde 1 hasta N
               Escribir "Ingrese el dato", I
               Leer VECTOR[I]
               Hacer I ←—— I + 1
     3.2 {Fin del ciclo del paso 3.1}
           Hacer I ←—— 1
     3.3 Repetir con I desde 1 hasta N – 1
                 Hacer MENOR ←—— VECTOR[I], K ←—— I y J ←—— I + 1
       3.3.1 Repetir con J desde I + 1 hasta N
         3.3.1.1 Si VECTOR[J] < MENOR entonces
                       Hacer MENOR ←—— VECTOR[J] y K ←—— J
         3.3.1.2 {Fin del condicional del paso 3.3.1.1}
                       Hacer J ←—— J + 1
       3.3.2 {Fin del ciclo del paso 3.3.1}
                 Hacer VECTOR[K] ←—— VECTOR[I], VECTOR[I] ←—— MENOR e
                       I ←—— I + 1
     3.4 {Fin del ciclo del paso 3.3}
           Escribir "Arreglo ordenado"
           Hacer I ←—— 1
     3.5 Repetir con I desde 1 hasta N
                 Escribir VECTOR[I]
                 Hacer I ←—— I + 1
     3.6 {Fin del ciclo del paso 3.5}
     sino
           Escribir "Error en los datos"
4. {Fin del condicional del paso 3}
```

www.FreeLibros.com

Problema 4.7

Búsqueda binaria

La búsqueda binaria consiste en dividir el intervalo de búsqueda en dos partes y comparar el elemento buscado con el elemento central del arreglo. En caso de ser diferentes, se deben redefinir los extremos del intervalo considerando si el elemento buscado es mayor o menor que el elemento que se encuentra en la posición central. De esta forma, se reduce el espacio de búsqueda. El proceso concluye cuando el elemento es encontrado o bien cuando el espacio de búsqueda se anula. Esto último implica que el elemento buscado no se encuentra en el arreglo.

Es importante destacar que el método funciona únicamente con arreglos ordenados. En cada iteración del método el espacio de búsqueda se reduce a la mitad, por lo que el número de comparaciones a realizar disminuye considerablemente. Esta disminución en el número de comparaciones será más notoria cuando más grande sea el tamaño del arreglo. A continuación presentamos el diagrama de flujo correspondiente.

Datos: VECTOR [1..N], X $\quad 1 \leq N \leq 50$

Donde:

VECTOR es un arreglo unidimensional de tipo entero.
X es una variable de tipo entero que representa el dato que se va a buscar.

Explicación de las variables

N: Variable de tipo entero. Indica el total de componentes que tendrá el arreglo unidimensional.
I: Variable de tipo entero. Se utiliza como índice del arreglo y como variable de control de diferentes ciclos.
VECTOR: Arreglo unidimensional de tipo entero.
X: Variable de tipo entero. Representa el dato que se va a buscar.
IZQ.: Variable de tipo entero. Se utiliza para almacenar el extremo izquierdo del intervalo.
DER.: Variable de tipo entero. Se utiliza para almacenar el extremo derecho del intervalo.
CEN.: Variable de tipo entero. Se utiliza para almacenar el centro del intervalo.
BAN.: Variable de tipo entero. Se utiliza para interrumpir el ciclo si se localiza el elemento buscado.

www.FreeLibros.com

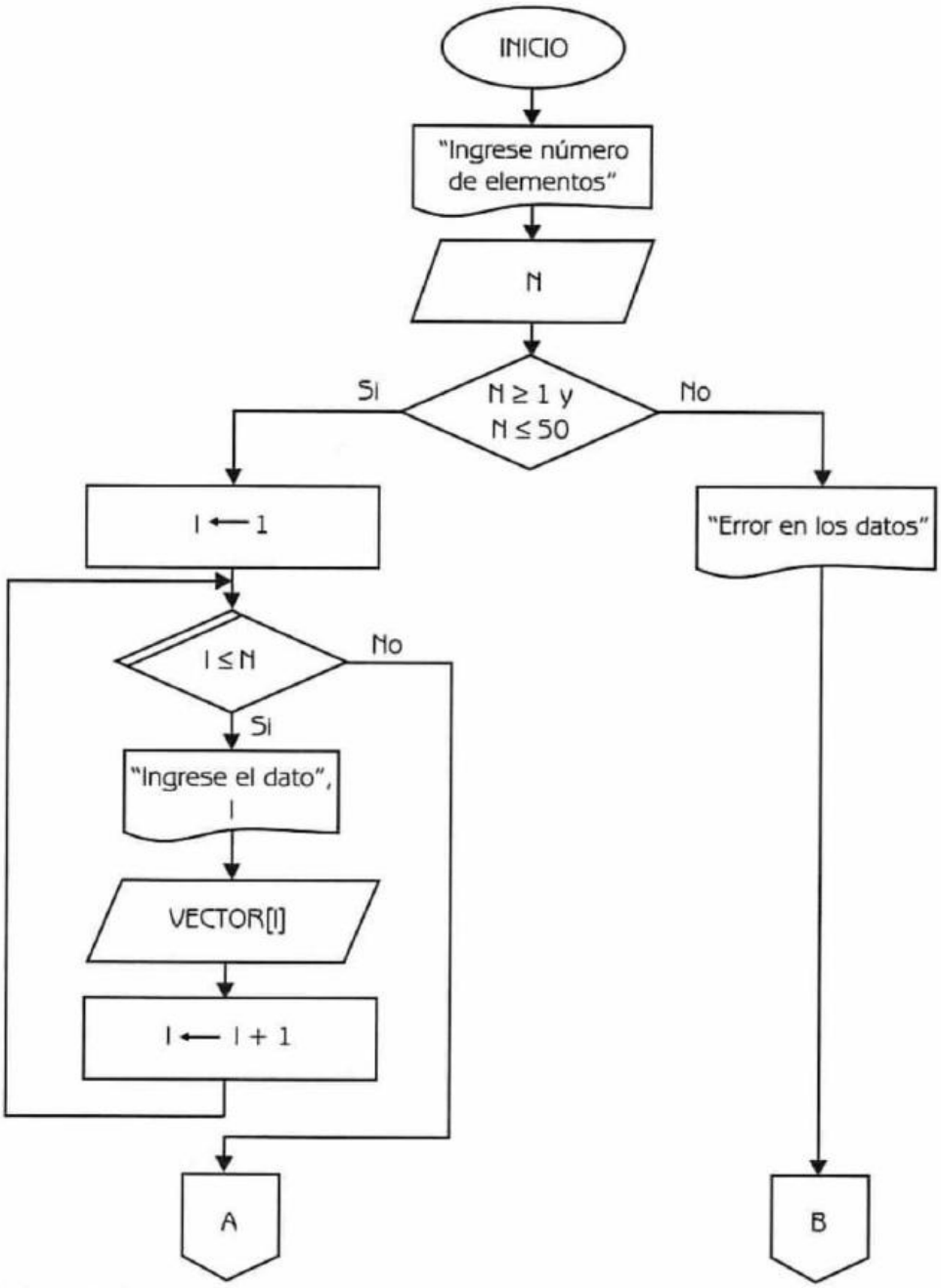


Diagrama de Flujo 4.15 (*continúa*)

www.FreeLibros.com

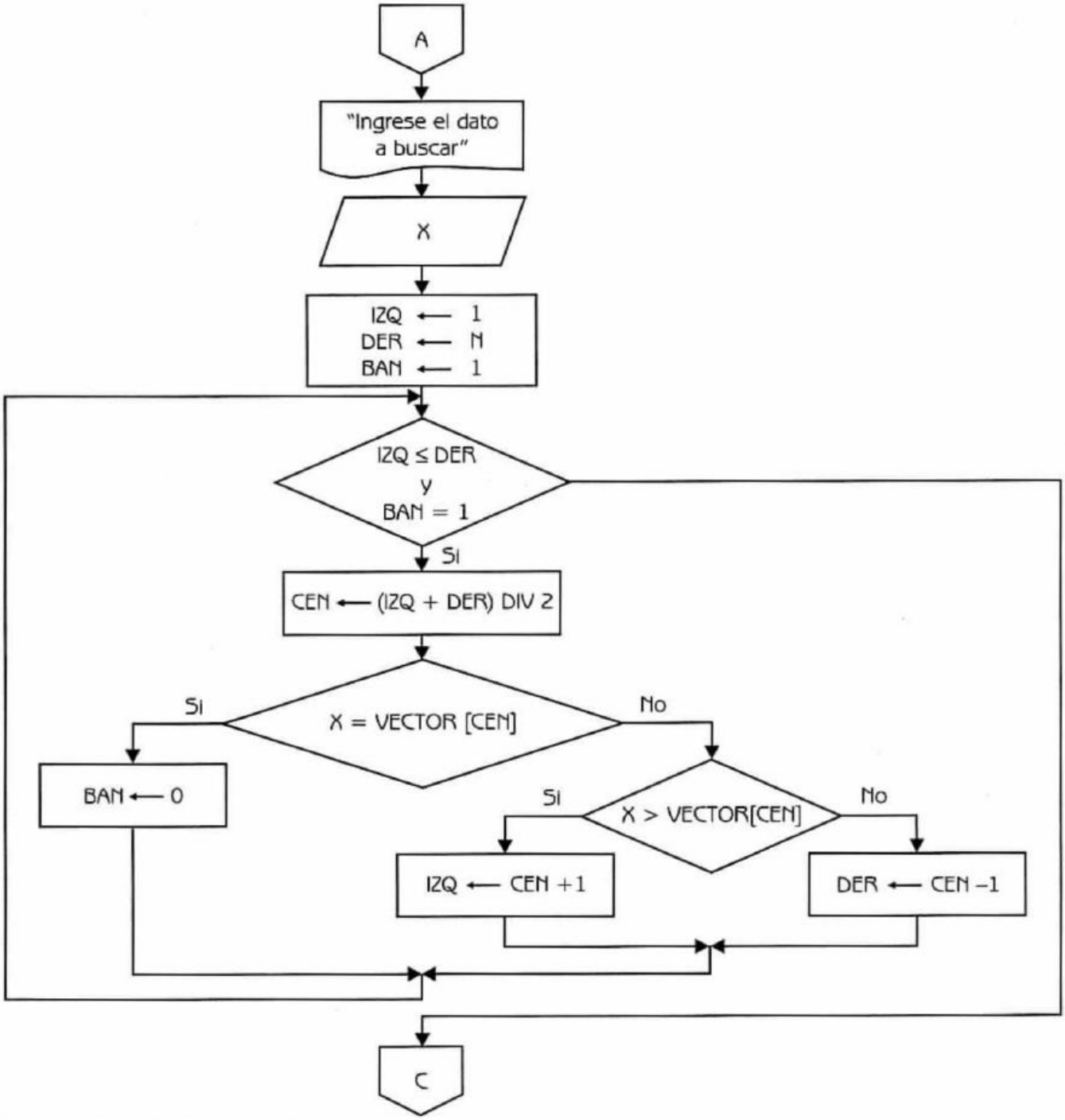


Diagrama de Flujo 4.15 (*continuación...*)

www.FreeLibros.com

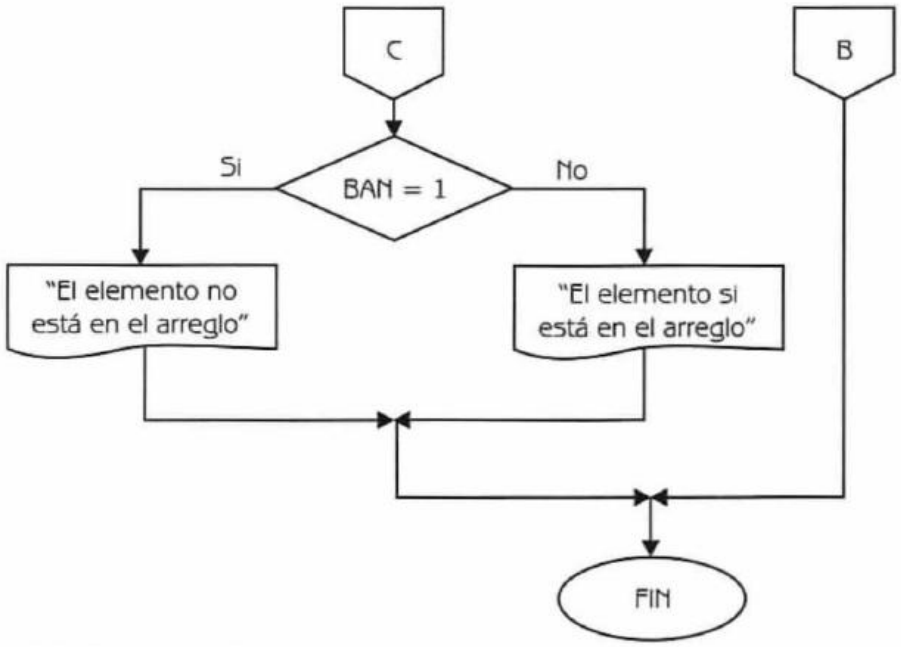


Diagrama de Flujo 4.15 (*continuación*)

A continuación presentamos el diagrama de flujo en lenguaje algorítmico.

Programa 4.15

```
Búsqueda binaria
{El programa realiza búsqueda binaria sobre un arreglo unidimensional ordenado}

{N, I, X, IZQ, DER, y CEN son variables de tipo entero. VECTOR es un arreglo unidimensional de tipo entero}

1. Escribir "Ingrese número de datos"
2. Leer N
3. Si (N >= 1) y (N <= 50)
      entonces
         Hacer I ⟵ 1
   3.1 Repetir con I desde 1 hasta N
            Escribir "Ingrese el dato", I
            Leer VECTOR [I]
            Hacer I ⟵ I + 1
   3.2 {Fin del ciclo del paso 3.1}
         Escribir "Ingresa el dato a buscar"
         Leer X
         Hacer IZQ ⟵ 1, DER ⟵ N y BAN ⟵ 1
```

www.FreeLibros.com

```
3.3 Mientras (IZQ <= DER) y (BAN = 1)
        Hacer CEN ←— (IZQ + DER) DIV 2
  3.3.1 Si (X = VECTOR [CEN])
        entonces
            Hacer BAN ←— 0
        sino
    3.3.1.1 Si X > VECTOR [CEN]
            entonces
                Hacer IZQ ←— CEN +1
            sino
                Hacer DER ←— CEN –1
    3.3.1.2 {Fin del condicional del paso 3.3.1.1}
  3.3.2 {Fin del condicional del paso 3.3.1}
3.4 {Fin del ciclo del paso 3.3}
3.5 Si (BAN = 1)
        entonces
            Escribir "El elemento no está en el arreglo"
        sino
            Escribir "El elemento sí está en el arreglo"
3.6 {Fin del condicional del paso 3.5}
 sino
        Escribir "Error en los datos"
4. {Fin condicional del paso 3}
```

En las siguientes tres tablas se puede observar el seguimiento del algoritmo para diferentes corridas.

Dato: VECTOR

5	12	14	18	21	33	38	45	52	65	76
1	2	3	4	5	6	7	8	9	10	11

Tabla 4.5

N	X	IZQ	DER	BAN	CEN	X = VECTOR [CEN]	X > VECTOR [CEN]	RESULTADO
11	65	1	11	1	6	65 = 33? NO	65 > 33? SI	
		7			9	65 = 52? NO	65 > 52? SI	
		10		0	10	65 = 65? SI		"EL elemento si está en el arreglo"

www.FreeLibros.com

Tabla 4.6								
N	X	IZQ	DER	BAN	CEN	X = VECTOR [CEN]	X > VECTOR [CEN]	RESULTADO
11	18	1	11	1	6	18 = 30? NO	18 > 33? NO	
			5		3	18 = 14? NO	18 > 14? SI	
		4		0	4	18 = 18? SI		"EL elemento si está en el arreglo"

Tabla 4.7								
N	X	IZQ	DER	BAN	CEN	X = VECTOR [CEN]	X > VECTOR [CEN]	RESULTADO
11	53	1	11	1	6	53=33? NO	53>33? SI	
		7			9	53=52? NO	53>52? SI	
		10			10	53=65? NO	53>65? NO	
			9					"EL elemento no está en el arreglo"

Problema 4.8

En un arreglo unidimensional se almacenan las calificaciones de N alumnos. Construya un diagrama de flujo que permita calcular e imprimir lo siguiente:

a) El promedio general del grupo.
b) Número de alumnos aprobados y número de alumnos reprobados (Si el alumno sacó una calificación menor a 6 se le considera reprobado).
c) Porcentaje de alumnos aprobados y reprobados.
d) Número de alumnos cuya calificación fue mayor a 8.

Dato: ALUMNO [1..N] $1 \leq N \leq 100$

Donde:

ALUMNO es un arreglo unidimensional de tipo real cuya capacidad máxima es de 100 alumnos.

www.FreeLibros.com

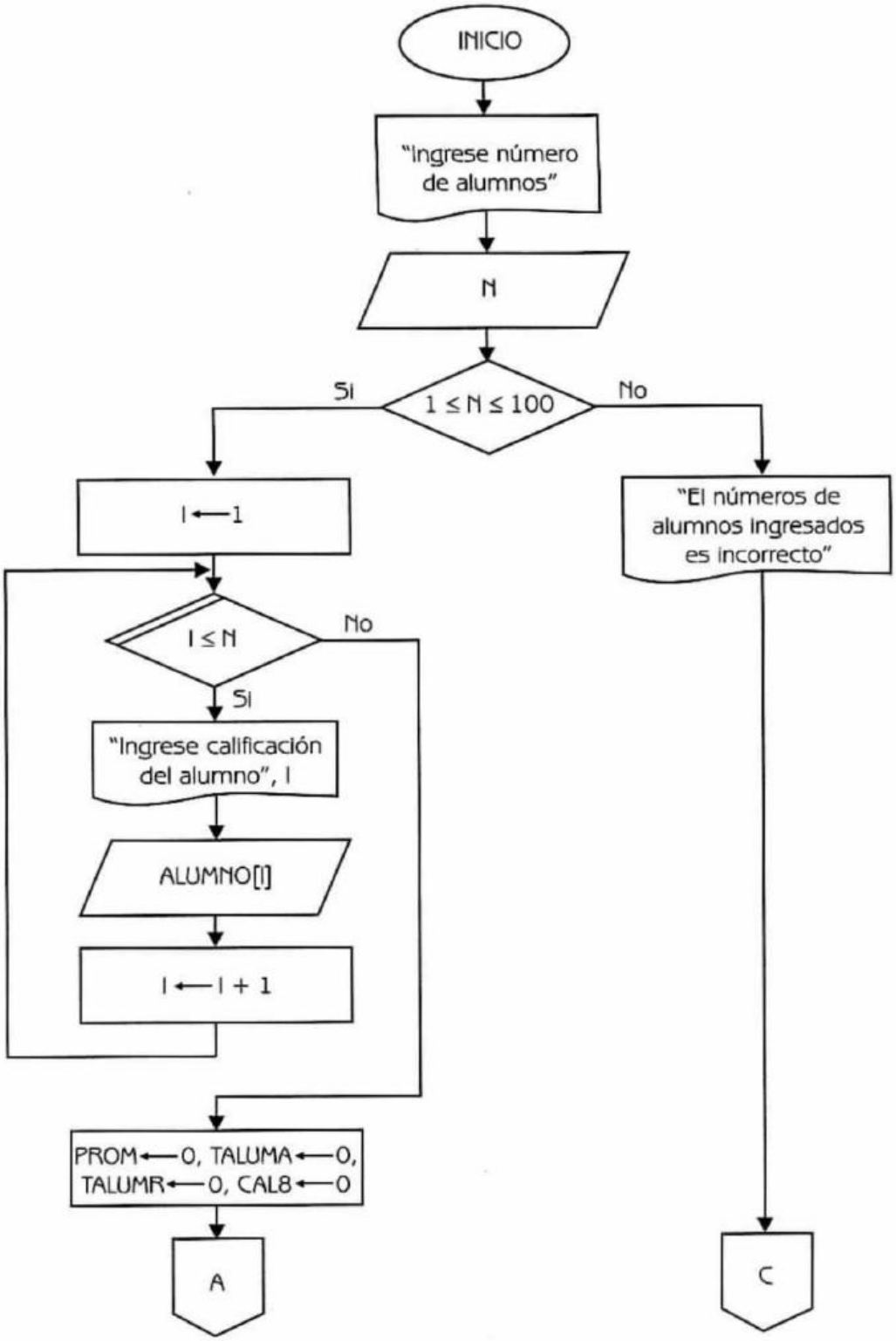


Diagrama de Flujo 4.16 (*continúa*)

www.FreeLibros.com

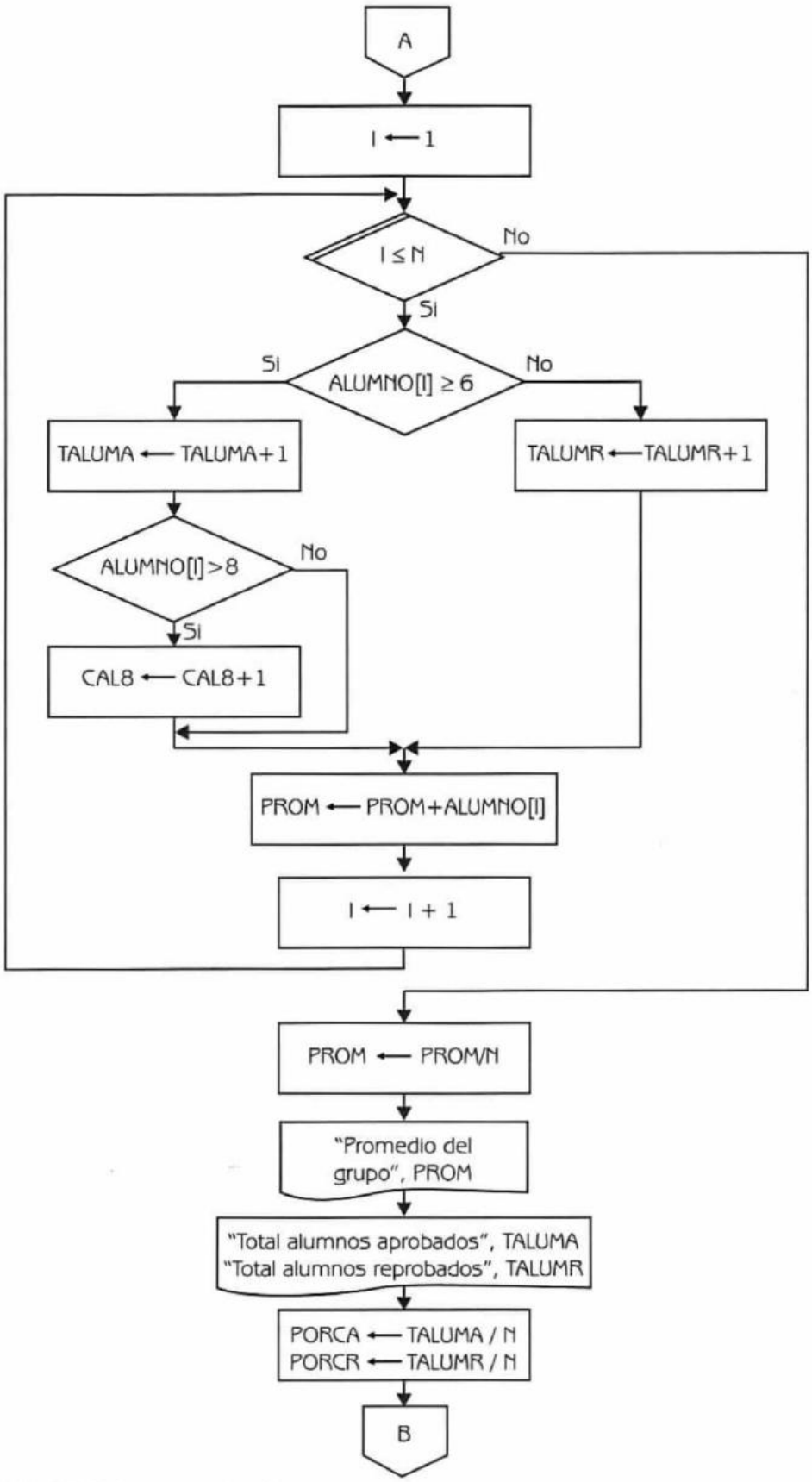


Diagrama de Flujo 4.16 (*continuación...*)

www.FreeLibros.com

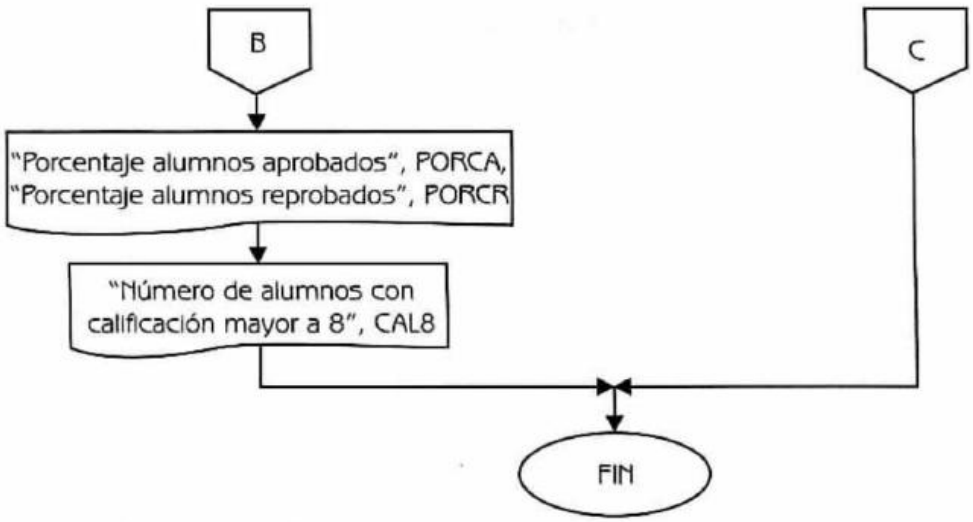


Diagrama de Flujo 4.16 (*continuación*)

Explicación de las variables

I: Variable de tipo entero. Se utiliza como variable de control de ciclos.

N: Variable de tipo entero. Almacena el número de alumnos. Su valor está comprendido en el intervalo [1..100].

ALUMNO: Arreglo unidimensional de tipo real.

PROM: Variable de tipo real. Se utiliza para acumular las calificaciones de los N alumnos, para posteriormente calcular el promedio de las mismas.

TALUMA: Variable de tipo entero. Se utiliza para contar el número de alumnos aprobados. También sirve para calcular el porcentaje de alumnos aprobados.

TALUMR: Variable de tipo entero. Se utiliza para contar el número de alumnos reprobados. También sirve para calcular el porcentaje de alumnos reprobados.

CAL8: Variable de tipo entero. Se utiliza para determinar el número de alumnos con calificación mayor a 8.

PORCA: Variable de tipo real. Almacena el porcentaje de alumnos aprobados.

PORCR: Variable de tipo real. Almacena el porcentaje de alumnos reprobados.

A continuación presentamos el programa correspondiente.

www.FreeLibros.com

Programa 4.16

```
CALIFICACIONES_GRUPO

{El programa, dadas las calificaciones de un grupo de N alumnos, obtiene: a) el
promedio general del grupo, b) el número de alumnos aprobados y reprobados, c)
el porcentaje de alumnos aprobados y reprobados, y d) el número de alumnos
cuya calificación fue mayor a 8}

{I, N, TALUMA, TALUMR y CAL8 son variables de tipo entero. PROM, PORCA y PORCR
son variables de tipo real. ALUMNO es un arreglo unidimensional de tipo real}

1. Escribir "Ingrese número de alumnos"
2. Leer N
3. Si (N ≥ 1) y (N ≤ 100)
        entonces
            Hacer I ←—— 1
    3.1 Repetir con I desde 1 hasta N
            Escribir "Ingrese calificaciones del alumno", I
            Leer ALUMNO[I]
            Hacer I ←——  I + 1
    3.2 {Fin del ciclo del paso 3.1}
        Hacer PROM ←—— 0, TALUMA ←—— 0, TALUMR ←—— 0 y CAL8 ←—— 0 e
                I ←—— 1
    3.3 Repetir con I desde 1 hasta N
        3.3.1 Si ALUMNO[I] ≥ 6
                    entonces
                        Hacer TALUMA ←—— TALUMA + 1
            3.3.1.1 Si ALUMNO[I] > 8 entonces
                            Hacer CAL8 ←—— CAL8 + 1
            3.3.1.2 {Fin del condicional del paso 3.3.1.1}
                    sino
                            Hacer TALUMR ←—— TALUMR + 1
        3.3.2 {Fin del condicional del paso 3.3.1}
            Hacer PROM ←—— PROM + ALUMNO[I] e I ←—— I + 1
    3.4 {Fin del ciclo del paso 3.3}
        Hacer PROM ←—— PROM / N
        Escribir "Promedio del grupo", PROM
        Escribir "Total alumnos aprobados", TALUMA
                 "Total alumnos reprobados", TALUMR
        Hacer PORCA ←—— TALUMA / N y PORCR ←—— TALUMR / N
        Escribir "Porcentaje alumnos aprobados", PORCA
                 "Porcentaje alumnos reprobados", PORCR
```

www.FreeLibros.com

```
            Escribir "Número de alumnos con calificación mayor a 8:", CAL8
        sino
            Escribir "El número de alumnos ingresado es incorrecto"
    4. {Fin del condicional del paso 3}
```

Problema 4.9

Dado un arreglo unidimensional de tipo entero que contiene las calificaciones de un grupo de alumnos que presentaron examen de ingreso para una universidad, construya un diagrama de flujo que calcule lo siguiente:

a) *La media aritmética*. Esta se calcula como la suma de los elementos entre el número de elementos.

b) *La varianza*. Esta se calcula como la suma de los cuadrados de las desviaciones de la media, entre el número de elementos.

c) *La desviación estándar*. Esta se calcula como la raíz cuadrada de la varianza.

d) *La moda*. Esta se calcula obteniendo el número que más frecuencia tiene.

Observemos el siguiente ejemplo:

Ejemplo 4.13

A	10	5	7	7	8	6	9	7	9	7	5	7

Media aritmética: 7.25
Varianza: 2.1875
Desviación estándar: 1.4790
Moda: 7

Dato: A[1..N] $1 \leq N \leq 50$

Donde: A es un arreglo unidimensional de tipo entero.

www.FreeLibros.com

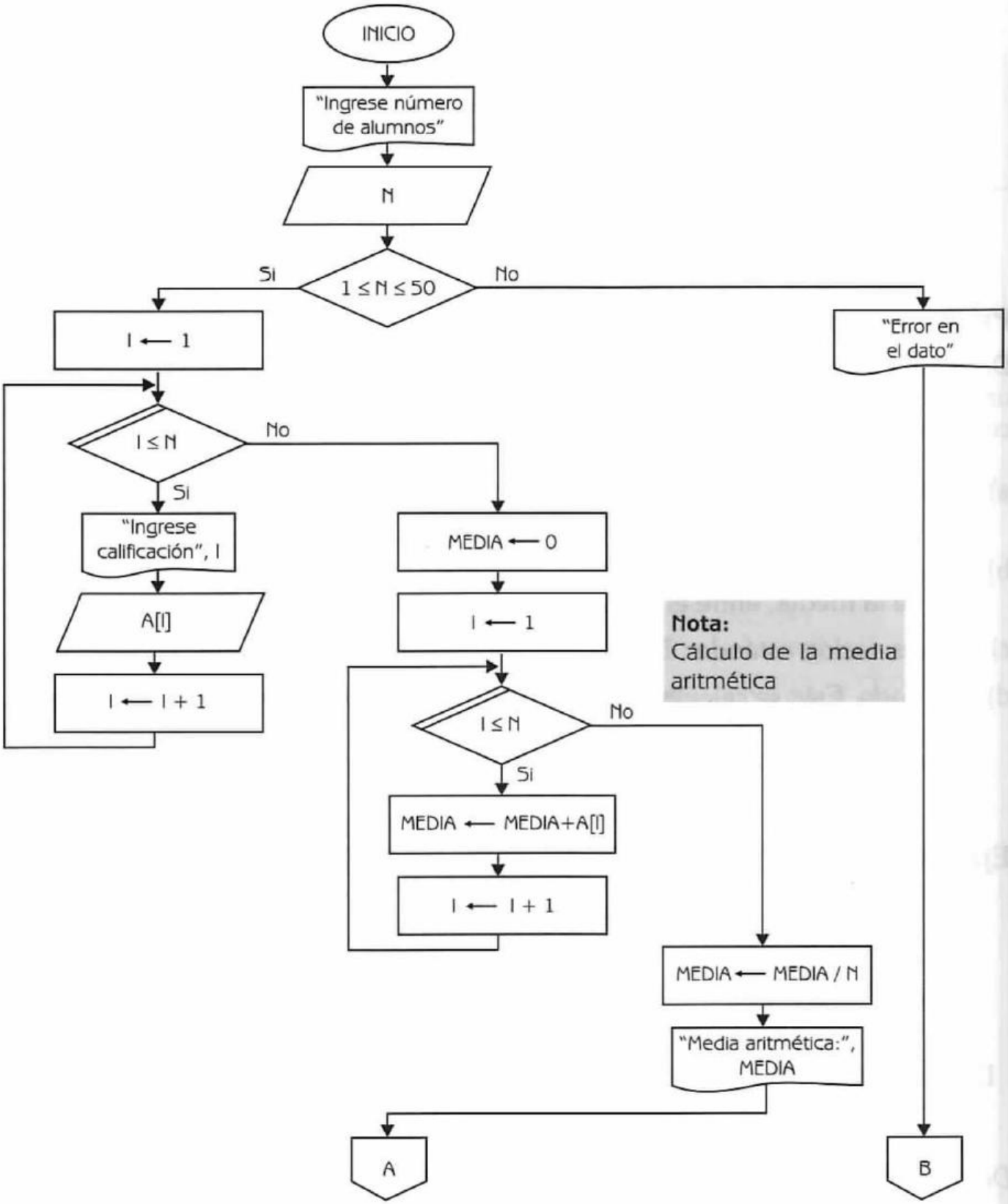
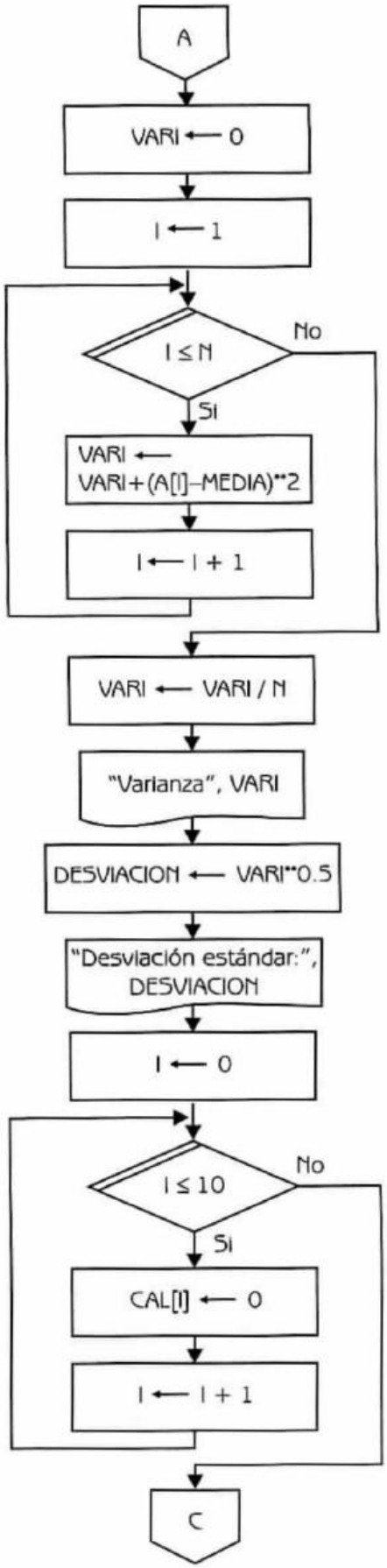


Diagrama de Flujo 4.17 (*continúa*)

www.FreeLibros.com



Nota:
Cálculo de la varianza.

Nota:
Cálculo de la desviación estándar.

Diagrama de Flujo 4.17 (*continuación...*)

www.FreeLibros.com

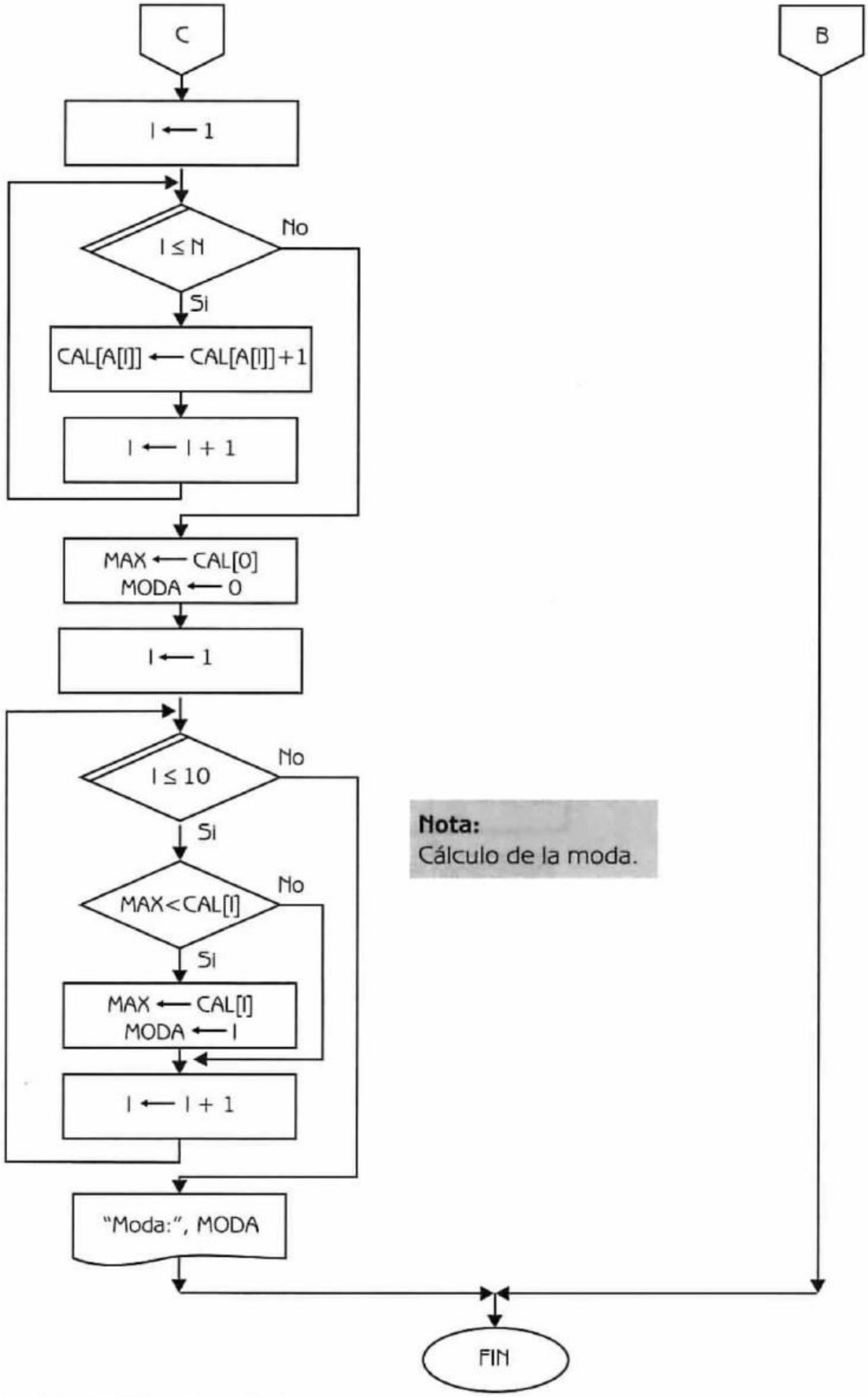


Diagrama de Flujo 4.17 (*continuación*)

www.FreeLibros.com

Explicación de las variables

N: Variable de tipo entero. Indica el número de alumnos que presentaron examen de admisión ($1 \leq N \leq 50$).
I: Variable de tipo entero. Se usa como variable de control en varios ciclos y como índice del arreglo.
A: Arreglo unidimensional de tipo entero.
MEDIA: Variable de tipo real. Se utiliza para calcular la media aritmética.
VARI: Variable de tipo real. Se usa para calcular la varianza.
DESVIACION: Variable de tipo real. Guarda la desviación estándar de los datos.
CAL: Arreglo unidimensional de tipo entero. CAL[0..10], almacena en cada posición I ($0 \leq I \leq 10$) el número de ocurrencias de la calificación I.
MAX: Variable de tipo entero. Se usa como auxiliar para calcular la moda de los datos.
MODA: Variable de tipo entero. Almacena la moda de los datos.

Programa 4.17

```
EXAMEN_INGRESO_UNIVERSIDAD

{El programa dado un arreglo unidimensional de tipo entero que contiene las calificaciones de un grupo de alumnos que presentaron examen de ingreso, calcula: a) la media aritmética, b) la varianza, c) la desviación estándar y d) la moda}

{N, I, MAX, y MODA son variables de tipo entero. MEDIA, VARI y DESVIACION son variables de tipo real. A y CAL son arreglos unidimensionales de tipo entero}

1. Escribir "Ingrese número de alumnos"
2. Leer N
3. Si (N ≥ 1) y (N ≤ 50)
        entonces
            Hacer I ←—— 1
   3.1 Repetir con I desde 1 hasta N
            Escribir "Ingrese calificación", I
            Leer A[I]
            Hacer I ←—— I + 1
   3.2 {Fin del ciclo del paso 3.1}
       Hacer MEDIA ←—— 0 e I ←—— 1
   3.3 Repetir con I desde 1 hasta N
```

www.FreeLibros.com

```
            Hacer MEDIA ⟵ MEDIA + A[I] e I ⟵ I + 1
    3.4 {Fin del ciclo del paso 3.3}
        Hacer MEDIA ⟵ MEDIA / N
        Escribir "Media aritmética", MEDIA
        Hacer VARI ⟵ 0 e I ⟵ 1
    3.5 Repetir con I desde 1 hasta N
            Hacer VARI ⟵ VARI + (A[I] – MEDIA) ** 2 e I ⟵ I + 1
    3.6 {Fin del ciclo del paso 3.5}
        Hacer VARI ⟵ VARI / N
        Escribir "Varianza", VARI
        Hacer DESVIACION ⟵ VARI ** 0.5
        Escribir "Desviación estándar: ",DESVIACION
        Hacer I ⟵ 0
    3.7 Repetir con I desde 0 hasta 10
            Hacer CAL[I] ⟵ 0 e I ⟵ I + 1
    3.8 {Fin del ciclo del paso 3.7}
        Hacer I ⟵ 1
    3.9 Repetir con I desde 1 hasta N
            Hacer CAL[ A[I] ] ⟵ CAL[ A[I] ] + 1 e I ⟵ I + 1
    3.10 {Fin del ciclo del paso 3.9}
         Hacer MAX ⟵ CAL[0], MODA ⟵ 0 e I ⟵ 1
    3.11 Repetir con I desde 1 hasta 10
      3.11.1 Si MAX < CAL[I] entonces
                  Hacer MAX ⟵ CAL [I] y MODA ⟵ I
      3.11.2 {Fin del condicional del paso 3.11.1}
             Hacer I ⟵ I + 1
    3.12 {Fin del ciclo del paso 3.11}
         Escribir "Moda:", MODA
       sino
          Escribir "Error en el dato"
4. {Fin del condicional del paso 3}
```

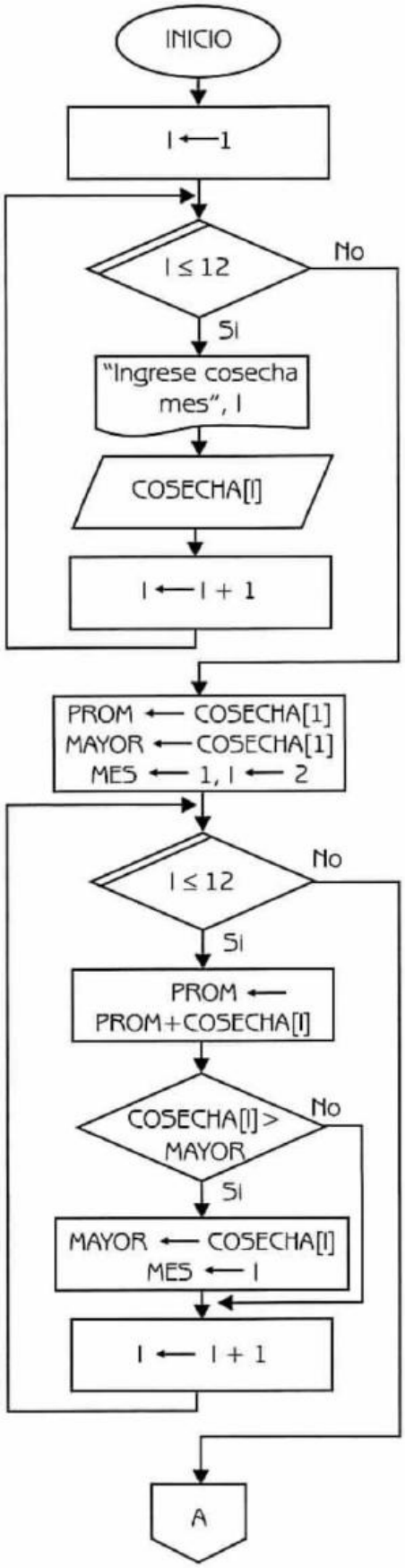
Problema 4.10

En un arreglo unidimensional se ha almacenado el número total de toneladas de cereales cosechadas durante cada mes del año anterior. Construya un diagrama de flujo que proporcione la siguiente información:

a) El promedio anual de toneladas cosechadas.
b) ¿Cuántos meses tuvieron una cosecha superior al promedio anual?
c) ¿Cuántos meses tuvieron una cosecha inferior al promedio anual?
d) ¿Cuál fue el mes en el que se produjeron mayor número de toneladas?

www.FreeLibros.com

Dato: COSECHA[1..12] (arreglo unidimensional de tipo real)



Nota:
En este ciclo se acumula para el cálculo del promedio, y además se busca el mes en el cual se cosecharon más toneladas de cereal.

Diagrama de Flujo 4.18 (*continúa*)

www.FreeLibros.com

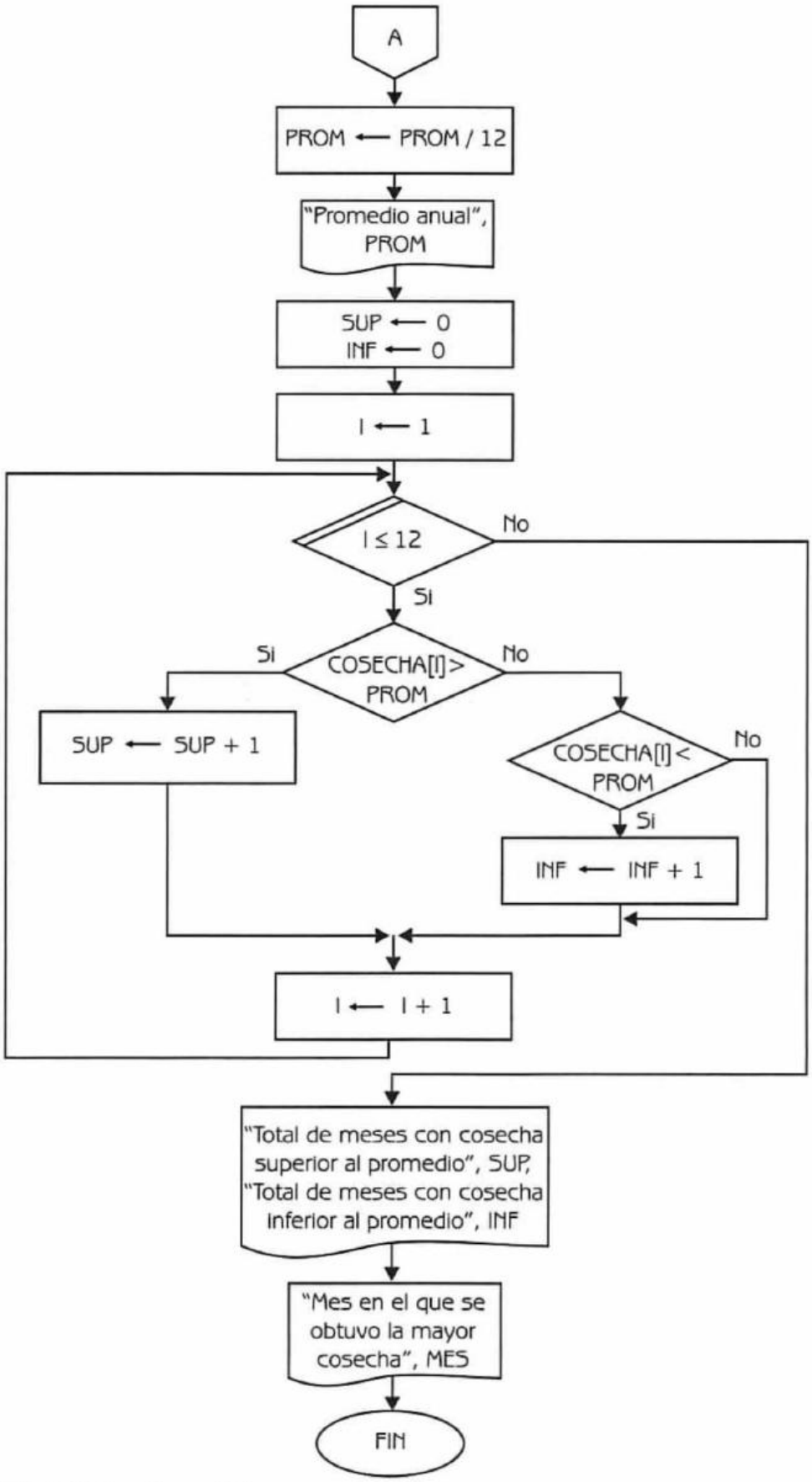


Diagrama de Flujo 4.18 (*continuación*)

www.FreeLibros.com

Explicación de las variables

I: Variable de tipo entero. Se utiliza como variable de control de ciclos.

COSECHA: Arreglo unidimensional de tipo real.

PROM: Variable de tipo real. Se utiliza para acumular las toneladas de cereales y calcular posteriormente el promedio anual.

MAYOR: Variable de tipo real. Almacena el total mensual más grande. Se utiliza para determinar el mes con la máxima cosecha.

MES: Variable de tipo entero. Almacena el mes en el cual se obtuvo la mayor cosecha.

SUP: Variable de tipo entero. Se utiliza para contar los meses que tuvieron cosecha superior al promedio.

INF: Variable de tipo entero. Se utiliza para contar los meses que tuvieron cosecha inferior al promedio.

A continuación presentamos el programa correspondiente.

Programa 4.18

```
COSECHA

{El programa, dada información en un arreglo unidimensional sobre las toneladas
de cereales cosechadas el mes anterior, obtiene valiosa información estadística}

{I, MES, SUP e INF son variables de tipo entero. PROM y MAYOR son variables de
tipo real. COSECHA es un arreglo unidimensional de tipo real}

1. Hacer I ⟵ 1
2. Repetir con I desde 1 hasta 12
        Escribir "Ingrese cosecha mes", I
        Leer COSECHA[I]
        Hacer I ⟵ I + 1
3. {Fin del ciclo del paso 2}
4. Hacer PROM ⟵ COSECHA[1], MAYOR ⟵ COSECHA[1], MES ⟵ 1 e
        I ⟵ 2
5. Repetir con I desde 2 hasta 12
        Hacer PROM ⟵ PROM + COSECHA[I]
     5.1 Si COSECHA[I] > MAYOR entonces
```

www.FreeLibros.com

```
                Hacer MAYOR ⟵ COSECHA[I] y MES ⟵ I
        5.2 {Fin del condicional del paso 5.1}
                Hacer I ⟵ I + 1
6. {Fin del ciclo del paso 5}
7. Hacer PROM ⟵ PROM / 12
8. Escribir "Promedio anual", PROM
9. Hacer SUP ⟵ 0, INF ⟵ 0 e I ⟵ 1
10. Repetir con I desde 1 hasta 12
      10.1 Si COSECHA[I] > PROM
                entonces
                    Hacer SUP ⟵ SUP + 1
                sino
          10.1.1 Si COSECHA[I] < PROM entonces
                    Hacer INF ⟵ INF + 1
          10.1.2 {Fin del condicional del paso 10.1.1}
      10.2 {Fin del condicional del paso 10.1}
           Hacer I ⟵ I + 1
11. {Fin del ciclo del paso 10}
12. Escribir "Total de meses con cosecha superior al promedio", SUP
            "Total de meses con cosecha inferior al promedio", INF
13. Escribir "Mes en el que se obtuvo la mayor cosecha", MES
```

Problema 4.11

Se tienen tres arreglos: SUR, CENTRO y NORTE que almacenan los nombres de los países del Sur, Centro y Norteamérica, respectivamente. Los tres arreglos están ordenados alfabéticamente.

Escriba un diagrama de flujo que mezcle los tres arreglos anteriores formando un cuarto arreglo, AMERICA, en el cual aparezcan los nombres de todos los países del continente ordenados alfabéticamente.

Datos: SUR[1..TPS], CENTRO[1..TPC], NORTE[1..TPN]

$$1 \leq TPS \leq 30 \qquad 1 \leq TPC \leq 30 \qquad 1 \leq TPN \leq 30$$

Donde:

SUR, CENTRO y NORTE	Son arreglos unidimensionales de tipo cadena de caracteres que almacenan los países de América del Sur, Centro y Norte respectivamente.

www.FreeLibros.com

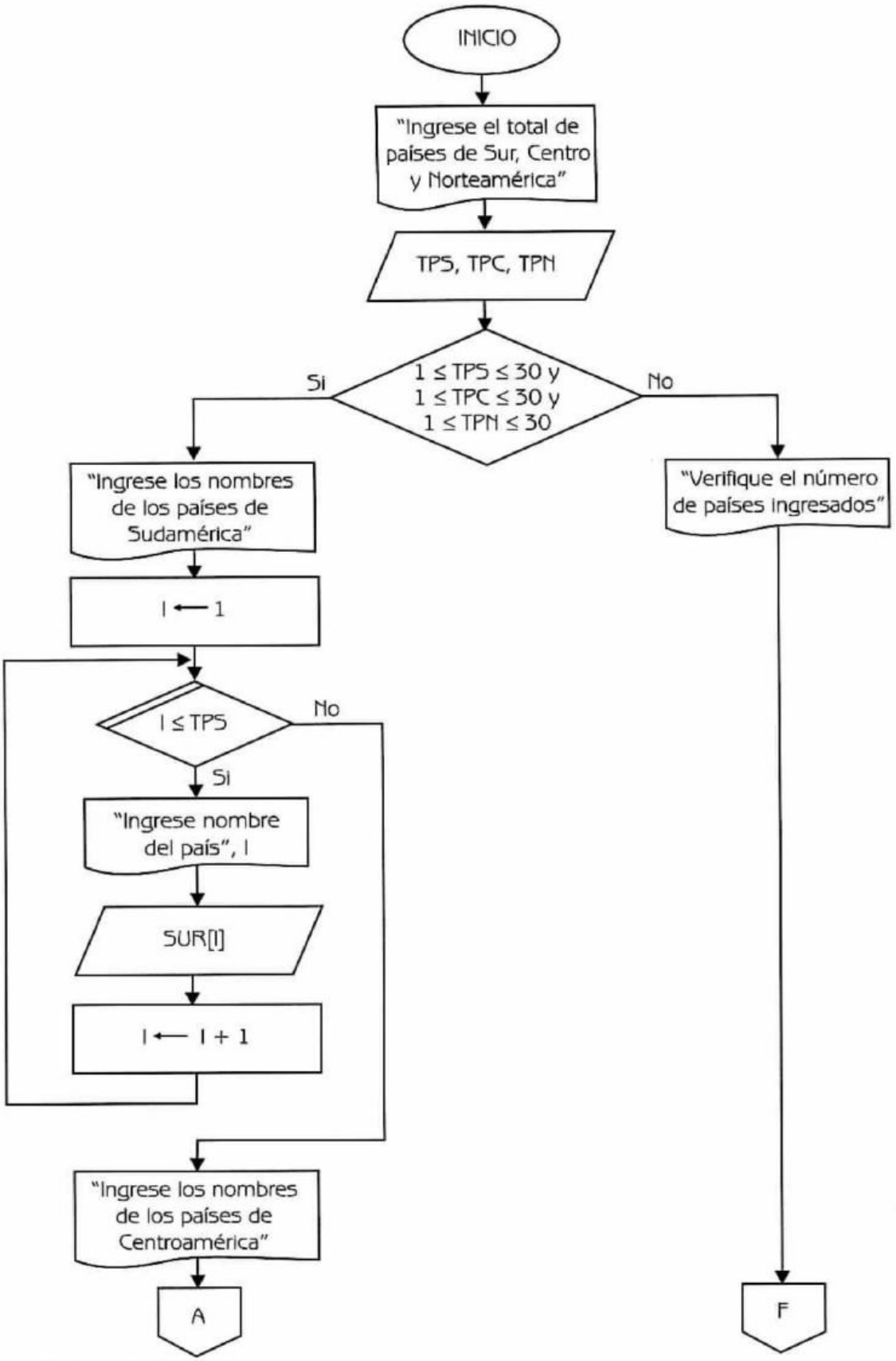


Diagrama de Flujo 4.19 (*continúa*)

www.FreeLibros.com

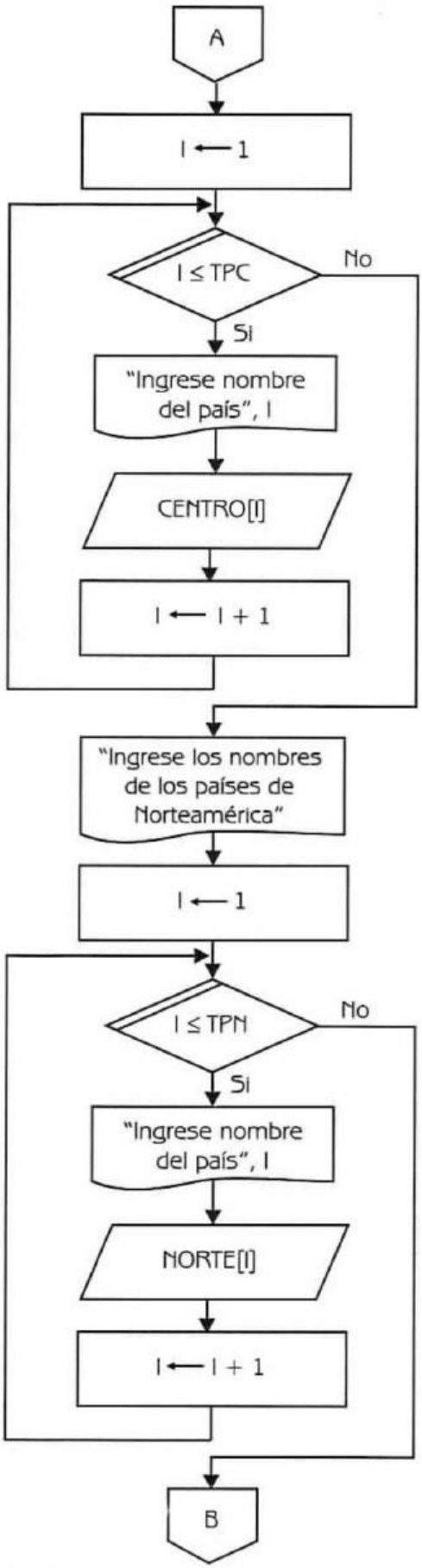


Diagrama de Flujo 4.19 (*continuación...*)

www.FreeLibros.com

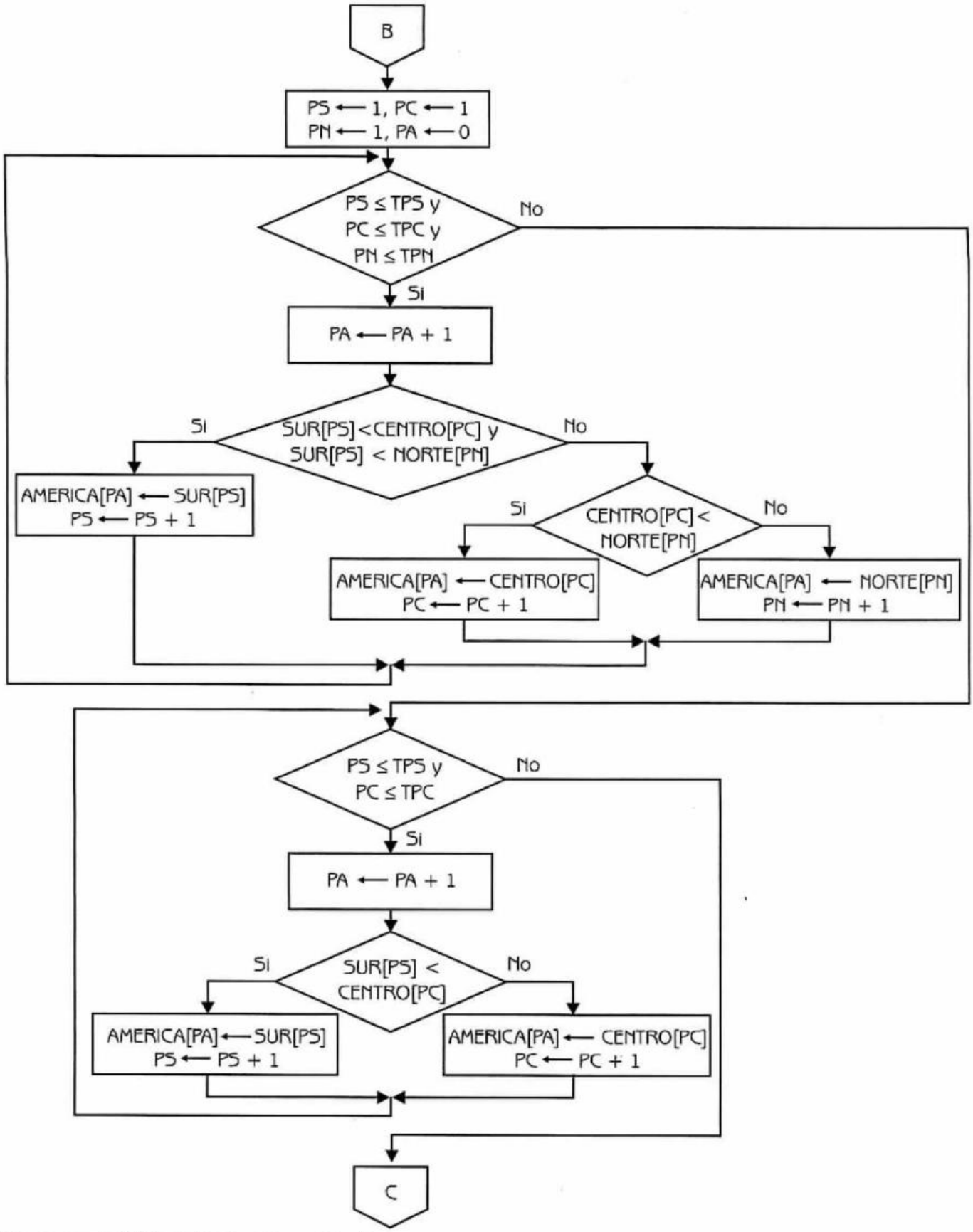


Diagrama de Flujo 4.19 *(continuación...)*

www.FreeLibros.com

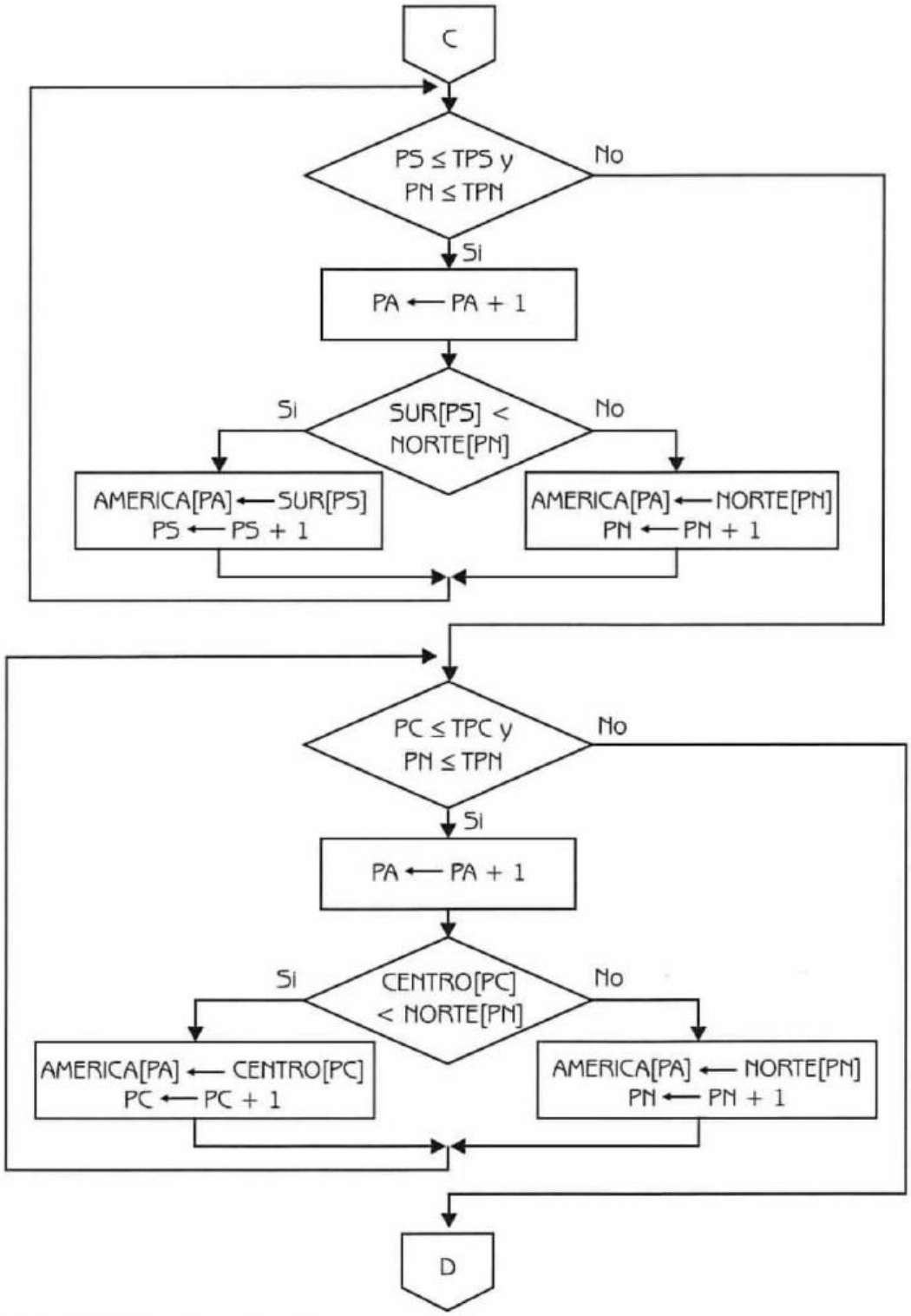


Diagrama de Flujo 4.19 (*continuación...*)

www.FreeLibros.com

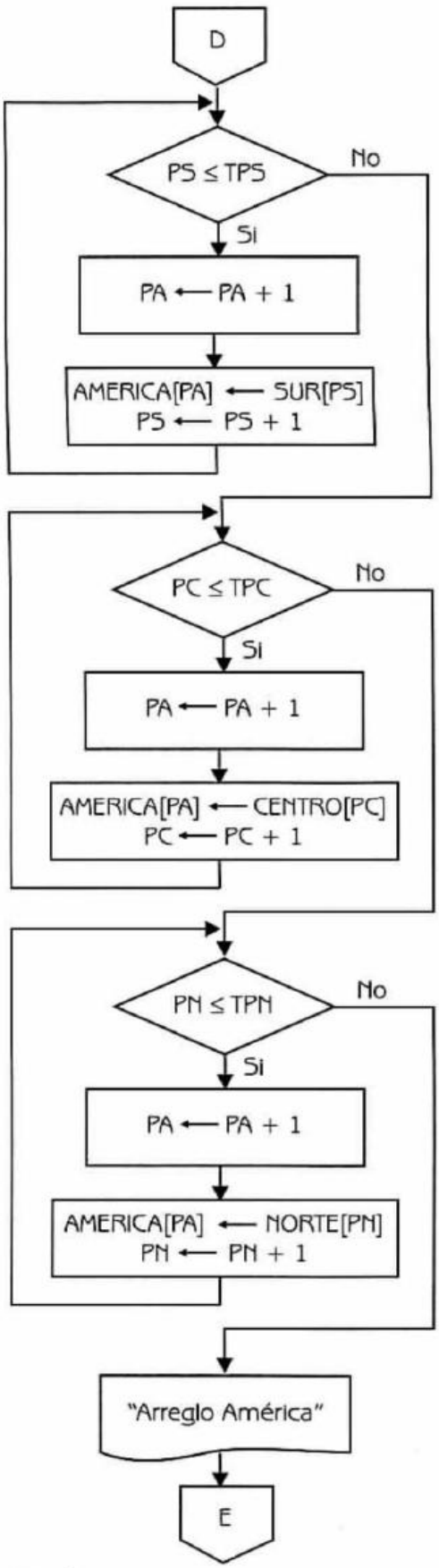


Diagrama de Flujo 4.19 *(continuación...)*

www.FreeLibros.com

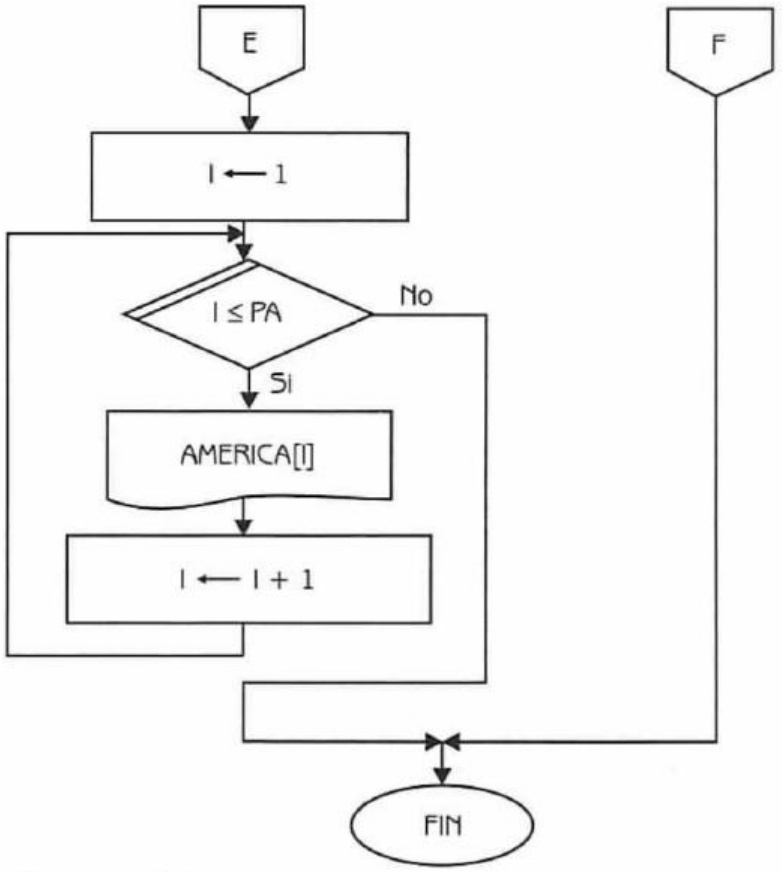


Diagrama de Flujo 4.19 *(continuación)*

Explicación de las variables

TPS: Variable de tipo entero. Indica el total de países de América del Sur. Su valor debe estar comprendido en el intervalo [1, 30].

TPC: Variable de tipo entero. Expresa el total de países de Centroamérica. Su valor debe estar comprendido en el intervalo [1, 30].

TPN: Variable de tipo entero. Indica el total de países de Norteamérica. Su valor debe estar comprendido en el intervalo [1, 30].

I: Variable de tipo entero. Se utiliza como variable de control en varios ciclos.

SUR, CENTRO y NORTE: Arreglos unidimensionales de tipo cadena de caracteres.

PS, PC, PN y PA: Variables de tipo entero. Se utilizan como índices de los arreglos unidimensionales, su valor se incrementa a medida que se asignan valores a los arreglos.

AMERICA: Arreglo unidimensional de tipo cadena de caracteres. Almacena los nombres de los países de América. Su capacidad máxima es para 90 países.

www.FreeLibros.com

A continuación presentamos el programa correspondiente.

Programa 4.19

```
PAISES_DE_AMERICA

{El programa, dados tres arreglos unidimensionales SUR, CENTRO y NORTE, que
contienen los países del Sur, Centro y Norteamérica, respectivamente, forma un
cuarto arreglo en el cual aparecen todos los países de América ordenados alfabéti-
camente}

{TPS, TPC, TPN, I, PS, PC, PN y PA son variables de tipo entero. SUR, CENTRO,
NORTE y AMERICA son arreglos unidimensionales de tipo cadena de caracteres}

1. Escribir "Ingrese el total de países de Sur, Centro y Norteamérica"
2. Leer TPS, TPC y TPN
3. Si (TPS >= 1 y TPS <= 30) y (TPC >= 1 y TPC <= 30) y (TPN >= 1
      y TPN <= 30)
   entonces
      Escribir "Ingrese los nombres de los países de Sudamérica"
      Hacer I ⟵ 1
   3.1 Repetir con I desde 1 hasta TPS
         Escribir "Ingrese nombre del país", I
         Leer SUR[I]
         Hacer I ⟵ I + 1
   3.2 {Fin del ciclo del paso 3.1}
      Escribir "Ingrese los nombres de los países de Centroamérica"
      Hacer I ⟵ 1
   3.3 Repetir con I desde 1 hasta TPC
         Escribir "Ingrese nombre del país", I
         Leer CENTRO[I]
         Hacer I ⟵ I + 1
   3.4 {Fin del ciclo del paso 3.3}
      Escribir "Ingrese los nombres de los países de Norteamérica"
      Hacer I ⟵ 1
   3.5 Repetir con I desde 1 hasta TPN
         Escribir "Ingrese nombre del país", I
         Leer NORTE[I]
         Hacer I ⟵ I + 1
   3.6 {Fin del ciclo del paso 3.5}
      Hacer PS ⟵ 1, PC ⟵ 1, PN ⟵ 1 y PA ⟵ 0
   3.7 Mientras (PS <= TPS) y (PC <= TPC) y (PN <= TPN) repetir
         Hacer PA ⟵ PA + 1
```

www.FreeLibros.com

```
    3.7.1 Si (SUR[PS] < CENTRO[PC]) y (SUR[PS] < NORTE[PN])
            entonces
                Hacer AMERICA[PA] ⟵ SUR [PS] y PS ⟵ PS + 1
            sino
      3.7.1.1 Si CENTRO[PC] < NORTE[PN]
                entonces
                    Hacer AMERICA[PA] ⟵ CENTRO [PC] y PC ⟵ PC + 1
                sino
                    Hacer AMERICA[PA] ⟵ NORTE [PN] y PN ⟵ PN + 1
        3.7.1.2 {Fin del condicional del paso 3.7.1.1}
    3.7.2 {Fin del condicional del paso 3.7.1}
  3.8 {Fin del ciclo del paso 3.7}
  3.9 Mientras (PS <= TPS) y (PC <= TPC) repetir
          Hacer PA ⟵ PA + 1
    3.9.1 Si SUR[PS] < CENTRO[PC]
            entonces
                Hacer AMERICA[PA] ⟵ SUR [PS] y PS ⟵ PS + 1
            sino
                Hacer AMERICA[PA] ⟵ CENTRO[PC] y PC ⟵ PC + 1
    3.9.2 {Fin del condicional del paso 3.9.1}
  3.10 {Fin del ciclo del paso 3.9}
  3.11 Mientras (PS <= TPS) y (PN <= TPN) repetir
          Hacer PA ⟵ PA + 1
    3.11.1 Si SUR[PS] < NORTE[PN]
            entonces
                Hacer AMERICA[PA] ⟵ SUR [PS] y PS ⟵ PS + 1
            sino
                Hacer AMERICA[PA] ⟵ NORTE [PN] y PN ⟵ PN + 1
    3.11.2 {Fin del condicional del paso 3.11.1}
  3.12 {Fin del ciclo del paso 3.11}
  3.13 Mientras (PC <= TPC) y (PN <= TPN) repetir
          Hacer PA ⟵ PA + 1
    3.13.1 Si CENTRO[PC] < NORTE[PN]
            entonces
                Hacer AMERICA[PA] ⟵ CENTRO [PC] y PC ⟵ PC + 1
            sino
                Hacer AMERICA[PA] ⟵ NORTE [PN] y PN ⟵ PN + 1
    3.13.2 {Fin del condicional del paso 3.13.1}
  3.14 {Fin del ciclo del paso 3.13}
  3.15 Mientras (PS <= TPS) repetir
          Hacer PA ⟵ PA + 1, AMERICA[PA] ⟵ SUR[PS] y
                PS ⟵ PS + 1
```

www.FreeLibros.com

```
    3.16 {Fin del ciclo del paso 3.15}
    3.17 Mientras (PC <= TPC) repetir
            Hacer PA ⟵ PA + 1, AMERICA[PA] ⟵ CENTRO[PC] y
                  PC ⟵ PC + 1
    3.18 {Fin del ciclo del paso 3.17}
    3.19 Mientras (PN <= TPN) repetir
            Hacer PA ⟵ PA + 1, AMERICA[PA] ⟵ NORTE[PN] y
                  PN ⟵ PN + 1
    3.20 {Fin del ciclo del paso 3.19}
         Escribir "Arreglo América"
         Hacer I ⟵ 1
    3.21 Repetir con I desde 1 hasta PA
            Escribir AMERICA[I]
            Hacer I ⟵ I + 1
    3.22 {Fin del ciclo del paso 3.21}
      sino
         Escribir "Verifique el número de países ingresados"
4. {Fin del condicional del paso 3}
```

Problema 4.12

Dada una matriz cuadrada A, construya un diagrama de flujo que permita determinar si dicha matriz es simétrica. Se considera que una matriz es simétrica si A[i,j] = A[j,i] y esto se cumple para todos los elementos i, j de la matriz.

Dato: A [1..N, 1..N] $1 \leq N \leq 50$

Donde: A es un arreglo bidimensional de tipo entero.

www.FreeLibros.com

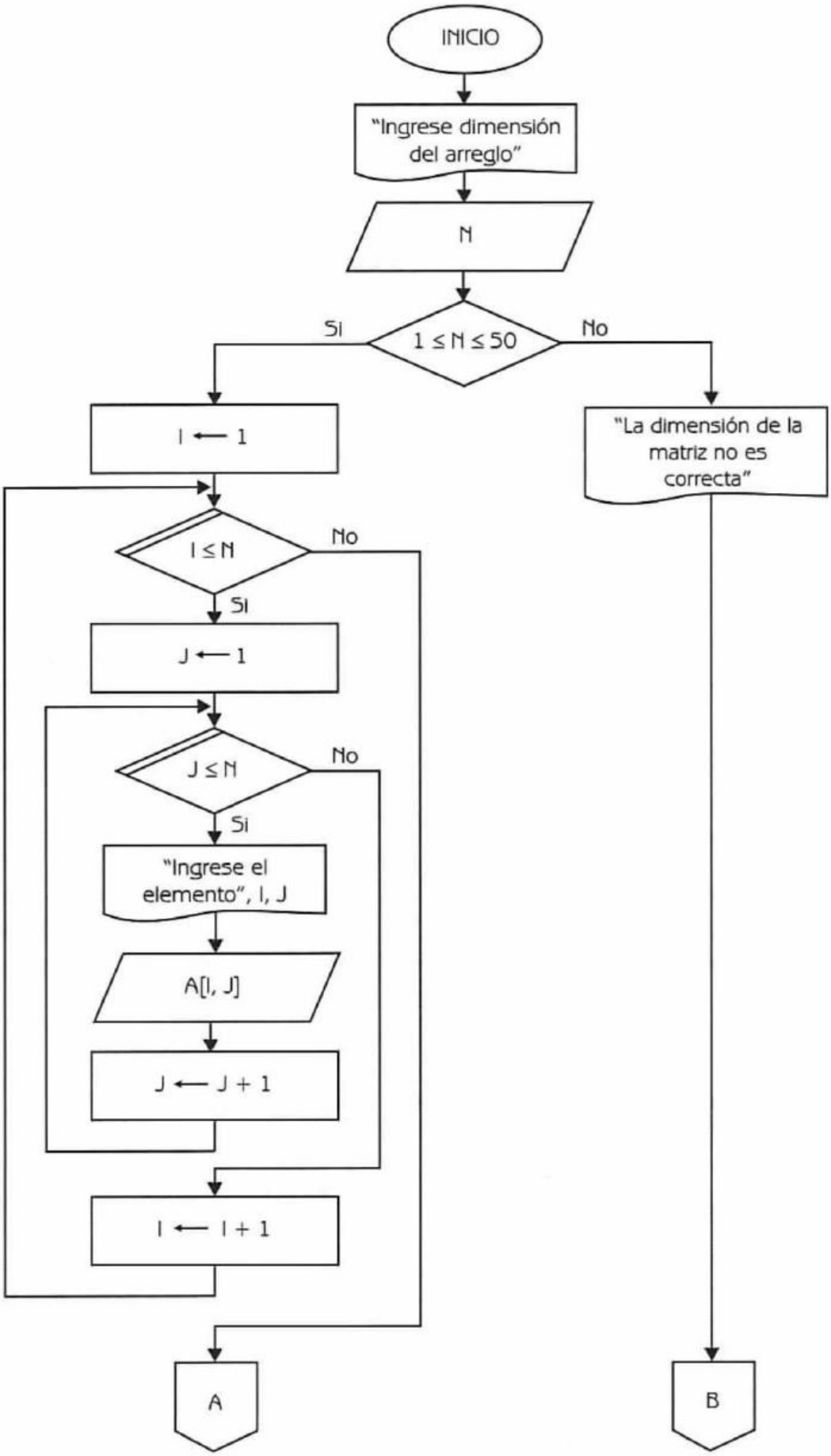


Diagrama de Flujo 4.20 (*continúa*)

www.FreeLibros.com

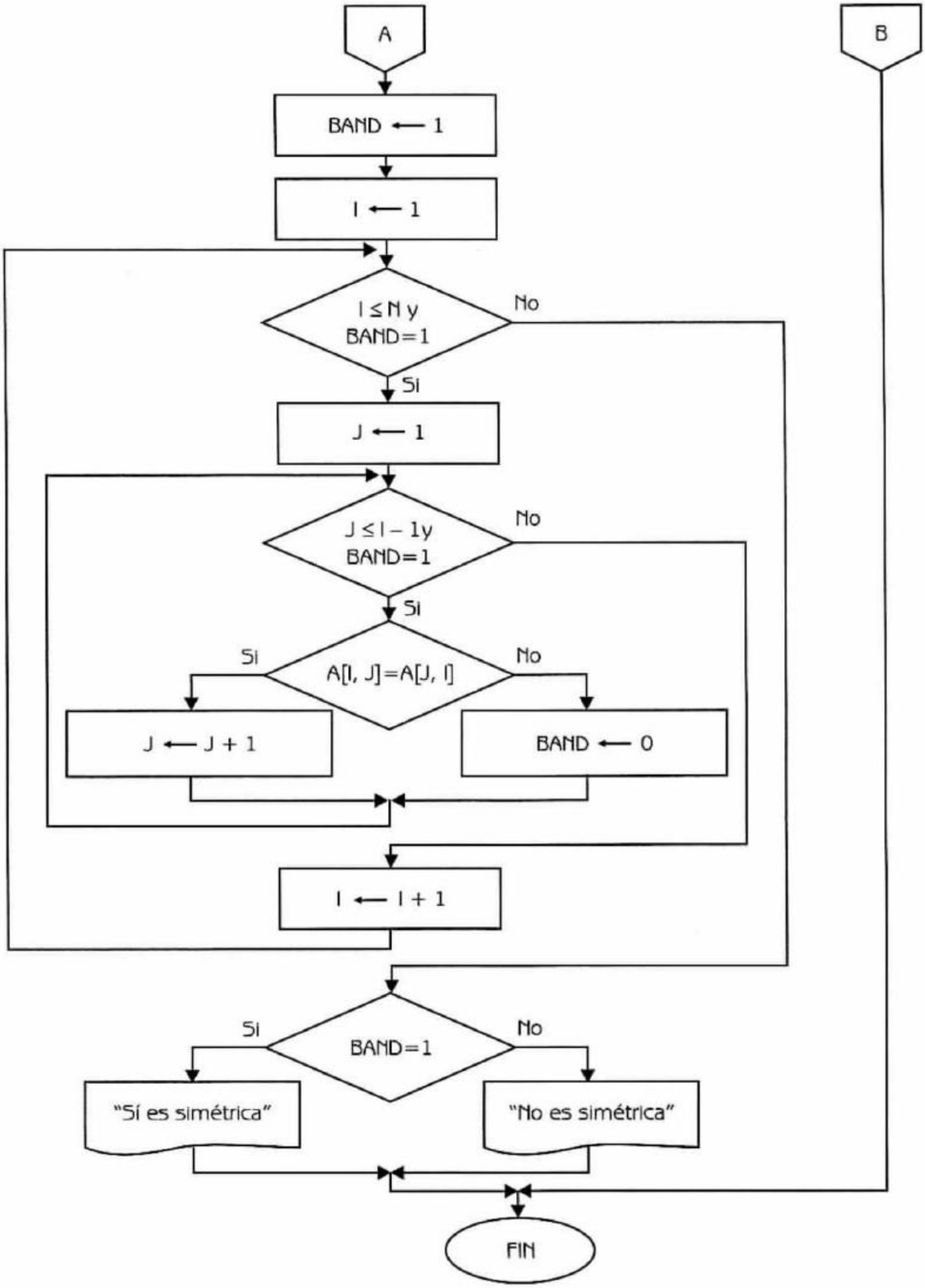


Diagrama de Flujo 4.20 (*continuación*)

www.FreeLibros.com

Explicación de las variables

N: Variable de tipo entero. Representa el número de renglones y de columnas del arreglo bidimensional.
I: Variable de tipo entero. Se usa como índice del arreglo.
J: Variable de tipo entero. Se usa como índice del arreglo.
A: Arreglo bidimensional de tipo entero.
BAND: Variable de tipo entero. Se utiliza para ahorrar iteraciones en el ciclo en el caso de que el elemento A[I,J] ≠ A[J,I], además sirve para indicar si la matriz es o no simétrica.

A continuación presentamos el programa correspondiente.

Programa 4.20

```
MATRIZ_SIMETRICA

{El programa, dada una matriz cuadrada, determina si es simétrica}

{N, I , J y BAND son variables de tipo entero. A es un arreglo bidimensional de tipo
entero}

1. Escribir "Ingrese dimensión del arreglo"
2. Leer N
3. Si (N >= 1) y (N <= 50)
          entonces
               Hacer I ←—— 1
          3.1 Repetir con I desde 1 hasta N
                    Hacer J ←—— 1
               3.1.1 Repetir con J desde 1 hasta N
                              Escribir "Ingrese el elemento", I, J
                              Leer A[I,J]
                              Hacer J ←—— J+1
               3.1.2 {Fin del ciclo del paso 3.1.1}
                    Hacer I ←—— I+1
          3.2 {Fin del ciclo del paso 3.1}
               Hacer BAND ←—— 1 e I ←—— 1
          3.3 Mientras (I <= N) y (BAND=1) repetir
                    Hacer J ←—— 1
               3.3.1 Mientras (J <= (I-1)) y (BAND=1) repetir
                    3.3.1.1 Si A[I,J] = A[J,I]
                                   entonces
                                        Hacer J ←—— J+1
```

www.FreeLibros.com

```
                                    sino
                                        Hacer BAND ←—— 0
                        3.3.1.2 {Fin del condicional del paso 3.3.1.1}
                    3.3.2 {Fin del ciclo del paso 3.3.1}
                        Hacer I ←—— I+1
                3.4 {Fin del ciclo del paso 3.3}
                3.5 Si BAND=1
                        entonces
                            Escribir "Sí es simétrica"
                        sino
                            Escribir "No es simétrica"
                3.6 {Fin del condicional del paso 3.5}
            sino
                Escribir "La dimensión de la matriz no es correcta"
    4. {Fin del condicional del paso 3}
```

Problema 4.13

Construya un diagrama de flujo que intercambie los renglones de un arreglo bidimensional. Los elementos del renglón 1 deben intercambiarse con los del renglón N, los del renglón 2 con los del renglón N - 1, y así sucesivamente. Por ejemplo, si A es:

A

	1	2	3	4
1	0	15	28	49
2	68	115	36	15
3	90	0	7	28
4	87	5	13	56

Luego del intercambio debe quedar:

A

	1	2	3	4
1	87	5	13	56
2	90	0	7	28
3	68	115	36	15
4	0	15	28	49

Dato: A[1..N, 1..N] $1 \leq N \leq 50$

Donde: A es un arreglo bidimensional de tipo entero.

www.FreeLibros.com

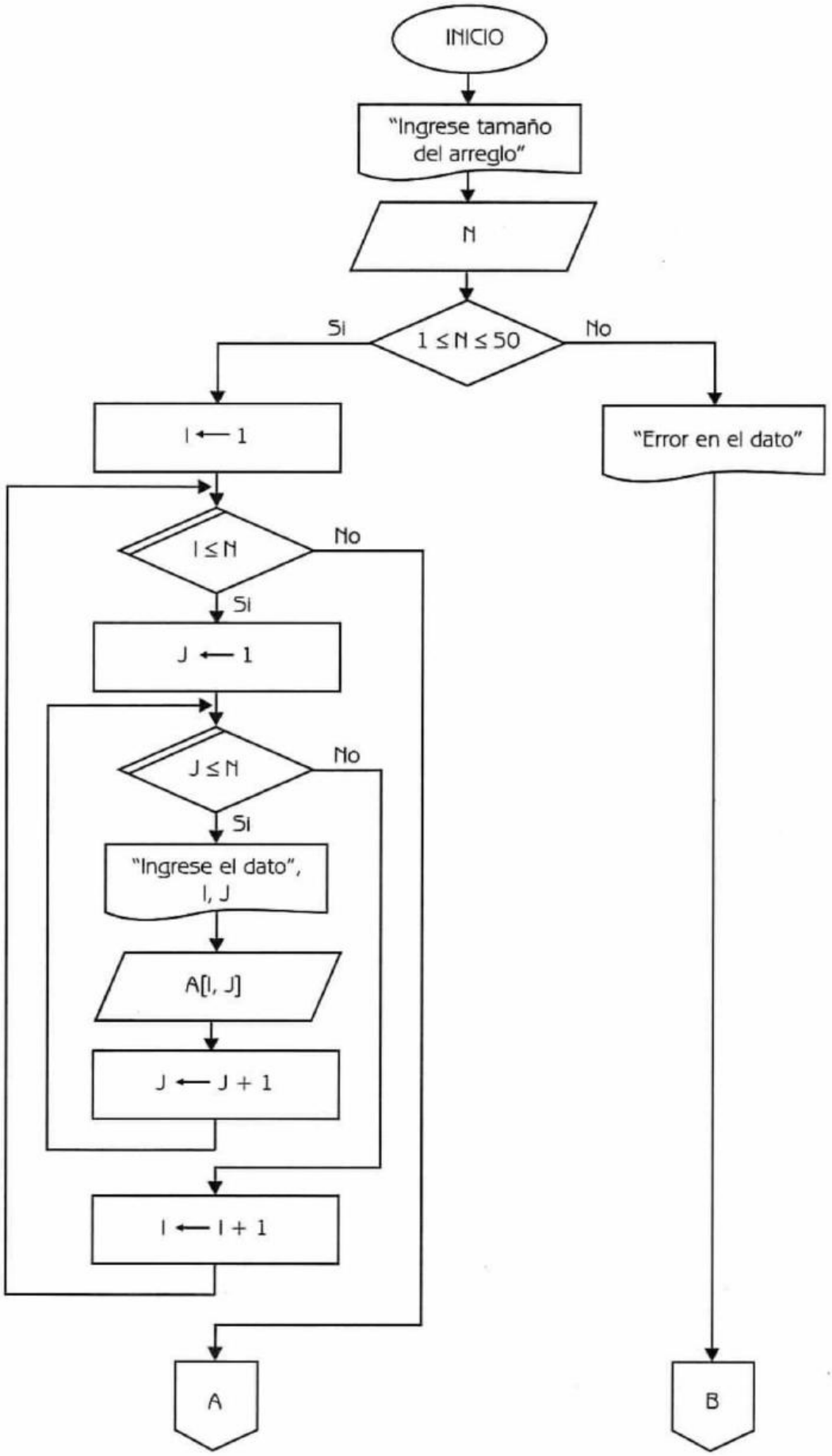


Diagrama de Flujo 4.21 (*continúa*)

www.FreeLibros.com

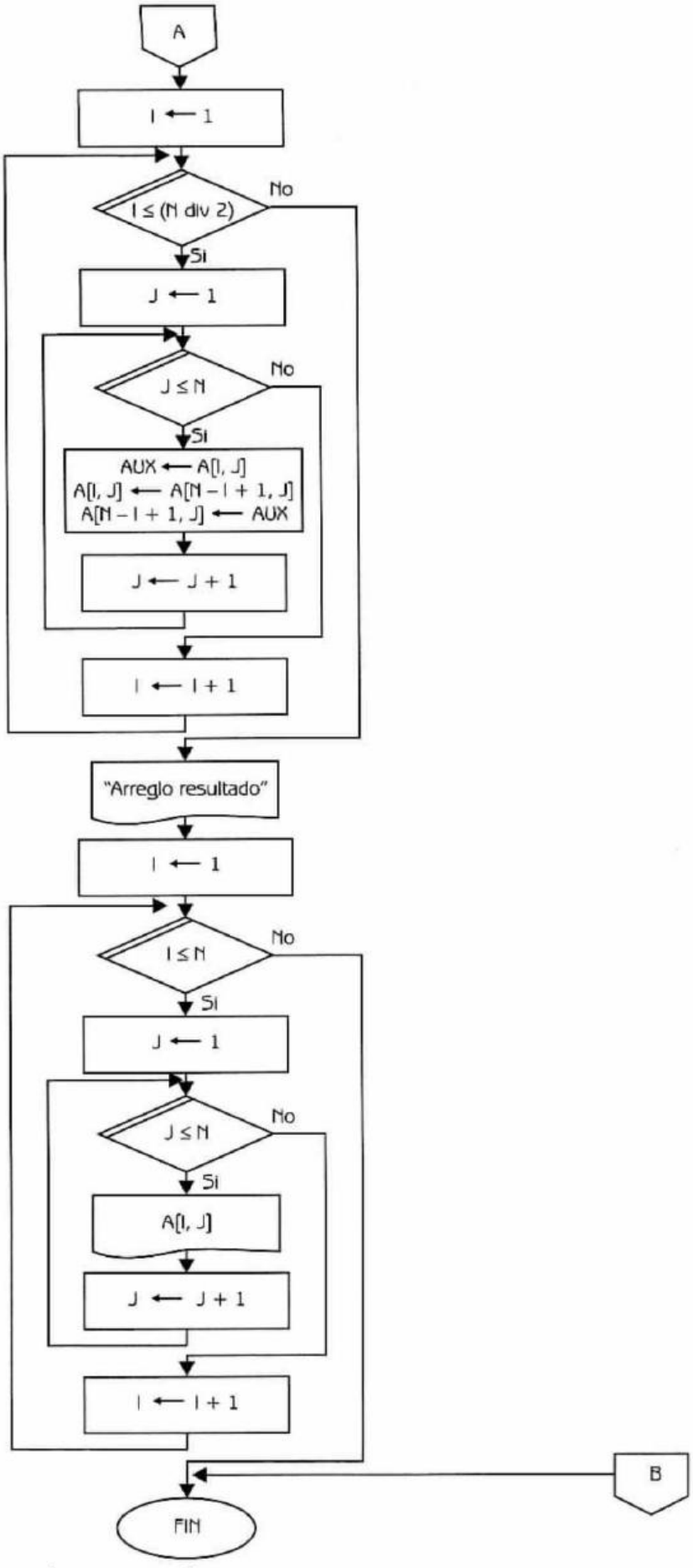


Diagrama de Flujo 4.21 (*continuación*)

www.FreeLibros.com

Explicación de las variables

N: Variable de tipo entero.

I: Variable de tipo entero. Se utiliza como variable de control de varios ciclos y como índice del arreglo.

J: Variable de tipo entero. Se utiliza como variable de control de varios ciclos y como índice del arreglo.

A: Arreglo bidimensional de tipo entero.

AUX: Variable de tipo entero. Se usa como auxiliar para realizar el intercambio de elementos del arreglo.

A continuación presentamos el programa correspondiente.

Programa 4.21

```
INTERCAMBIA_RENGLONES_MATRIZ

{El programa, dado un arreglo bidimensional cuadrado de tipo entero, intercambia
los renglones del mismo}

{I, N, J y AUX son variables de tipo entero. A es un arreglo bidimensional de tipo en-
tero}

1. Escribir "Ingrese tamaño del arreglo"
2. Leer N
3. Si (N >= 1 y N <= 50)
        entonces
              Hacer I ⟵ 1
        3.1 Repetir con I desde 1 hasta N
                    Hacer J ⟵ 1
          3.1.1 Repetir con J desde 1 hasta N
                          Escribir "Ingrese el dato", I, J
                          Leer A[I,J]
                          Hacer J ⟵ J + 1
          3.1.2 {Fin del ciclo del paso 3.1.1}
                    Hacer I ⟵ I + 1
        3.2 {Fin del ciclo del paso 3.1}
             Hacer I ⟵ 1
        3.3 Repetir con I desde 1 hasta (N div 2)
                  Hacer J ⟵ 1
          3.3.1 Repetir con J desde 1 hasta N
                              Hacer AUX ⟵ A[I,J], A[I,J] ⟵ A[N - I + 1,J],
                                    A[N - I + 1,J] ⟵ AUX y J ⟵ J + 1
```

www.FreeLibros.com

```
            3.3.2 {Fin del ciclo del paso 3.3.1}
                  Hacer I ←—— I + 1
        3.4 {Fin del ciclo del paso 3.3}
            Escribir "Arreglo resultado"
            Hacer I ←—— 1
        3.5 Repetir con I desde 1 hasta N
                  Hacer J ←—— 1
            3.5.1 Repetir con J desde 1 hasta N
                        Escribir A[I,J]
                        Hacer J ←—— J + 1
            3.5.2 {Fin del ciclo del paso 3.5.1}
                  Hacer I ←—— I + 1
        3.6 {Fin del ciclo del paso 3.5}
        sino
            Escribir "Error en el dato"
4. {Fin del condicional del paso 3}
```

Problema 4.14

Se tienen los costos de producción de tres departamentos (dulces, bebidas y conservas), correspondientes a los 12 meses del año anterior.

Construya un diagrama de flujo que pueda proporcionar la siguiente información:

a) ¿En qué mes se registró el mayor costo de producción de dulces?

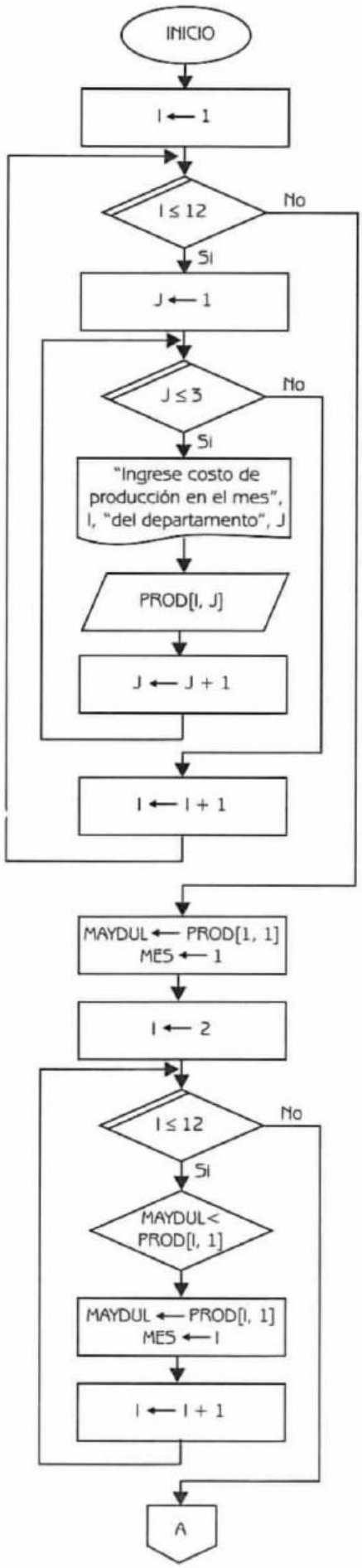
b) Promedio anual de los costos de producción de bebidas.

c) ¿En qué mes se registró el mayor costo de producción en bebidas, y en qué mes el menor costo?

d) ¿Cuál fue el rubro que tuvo el menor costo de producción en diciembre?

Dato: PROD [1..12,1..3] (arreglo bidimensional de tipo real que almacena los costos de producción de tres departamentos en los 12 meses del año anterior).

www.FreeLibros.com

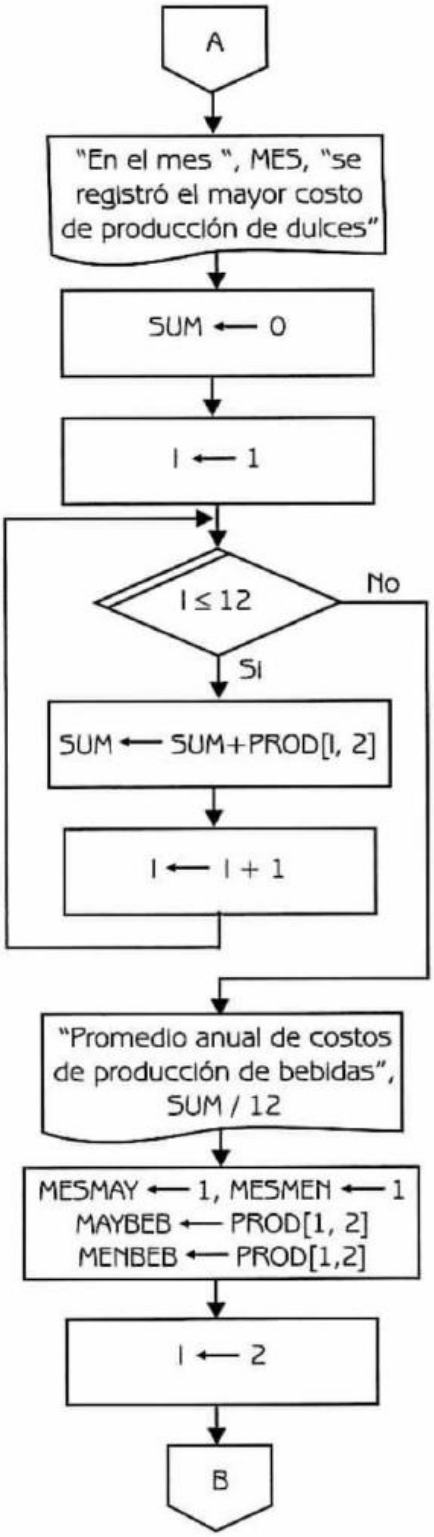


Nota:
Lectura de los costos de producción de los tres departamentos, en los 12 meses del año anterior.

Nota:
Cálculo del inciso A.

Diagrama de Flujo 4.22 (*continúa*)

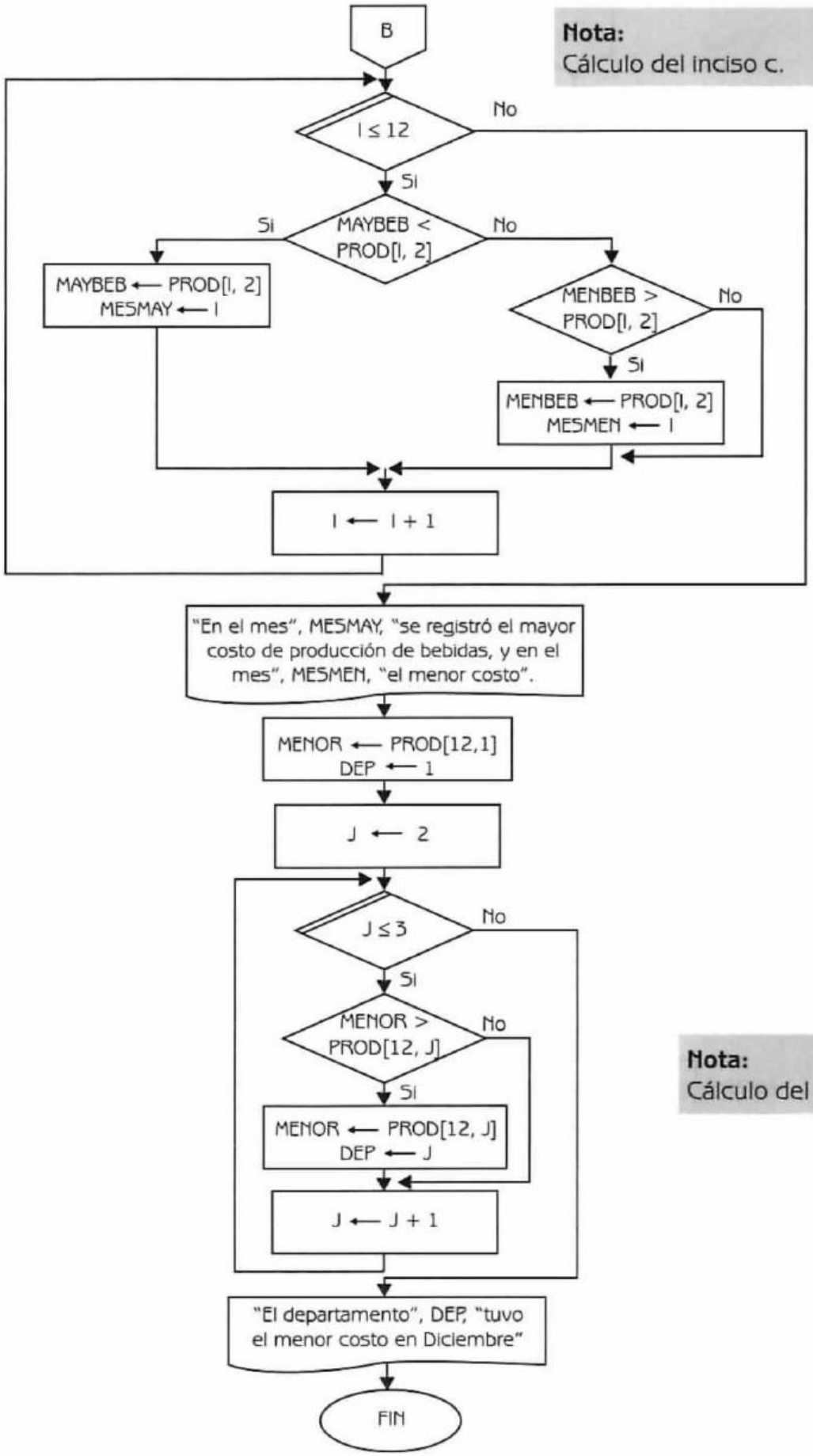
www.FreeLibros.com



Nota:
Cálculo del inciso b

Diagrama de Flujo 4.22 (*continuación...*)

www.FreeLibros.com



Nota:
Cálculo del inciso c.

Nota:
Cálculo del inciso d.

Diagrama de Flujo 4.22 (*continuación*)

www.FreeLibros.com

Explicación de las variables

I y *J*: Variables de tipo entero. Se usan como variables de control de varios ciclos y como índices de los arreglos.

PROD: Arreglo bidimensional de tipo real.

MAYDUL: Variable de tipo real. Se utiliza para almacenar el mayor costo de producción de dulces a lo largo del año.

MES: Variable de tipo entero. Almacena el número de mes en el cual se registró el mayor costo de producción de dulces.

SUM: Variable de tipo real. Se usa para sumar los costos mensuales de producción de bebidas, y para posteriormente calcular el promedio anual.

MESMAY: Variable de tipo entero. Se utiliza para almacenar el mes en el cual se registró el mayor costo de producción de bebidas.

MESMEN: Variable de tipo entero. Guarda el mes en el cual se registró el menor costo de producción de bebidas.

MAYBEB: Variable de tipo real. Almacena el mayor costo mensual de producción de bebidas.

MENBEB: Variable de tipo real. Almacena el menor costo mensual de producción de bebidas.

MENOR: Variable de tipo real. Se usa para almacenar el menor costo de producción en el mes de diciembre.

DEP: Variable de tipo entero. Almacena el número de departamento en el cual se registró el menor costo de producción en el mes de diciembre.

A continuación presentamos el diagrama de flujo en lenguaje algorítmico.

www.FreeLibros.com

Programa 4.22

```
PRODUCCION_DEPARTAMENTOS

{El programa, dados los costos de producción de tres departamentos de una fábrica
correspondientes a los 12 meses del año anterior, obtiene información útil para la
toma de decisiones de la empresa}

{I, J, MES, MESMAY, MESMEN y DEP son variables de tipo entero. MAYDUL, SUM,
MAYBEB, MENBEB y MENOR son variables de tipo real. PROD es un arreglo bidimen-
sional de tipo real}

1. Hacer I ⟵ 1
2. Repetir con I desde 1 hasta 12
        Hacer J ⟵ 1
   2.1 Repetir con J desde 1 hasta 3
       Escribir "Ingrese costo de producción en el mes", I, "del departamento", J
           Leer PROD[I,J]
           Hacer J ⟵ J+1
   2.2 {Fin del ciclo del paso 2.1}
       Hacer I ⟵ I+1
3. {Fin del ciclo del paso 2}
4. Hacer MAYDUL ⟵ PROD[1,1], MES ⟵ 1 e I ⟵ 2
5. Repetir con I desde 2 hasta 12
   5.1 Si MAYDUL < PROD[I,1] entonces
                Hacer MAYDUL ⟵ PROD[I,1] y MES ⟵ I
   5.2 {Fin del condicional del paso 5.1}
       Hacer I ⟵ I+1
6. {Fin del ciclo del paso 5}
7. Escribir "En el mes", MES, "se registró el mayor costo de producción de dulces"
8. Hacer SUM ⟵ 0 e I ⟵ 1
9. Repetir con I desde 1 hasta 12
        Hacer SUM ⟵ SUM + PROD[I,2] e I ⟵ I+1
10. {Fin del ciclo del paso 9}
11. Escribir "Promedio anual de costos de producción de bebidas", SUM/12
12. Hacer MESMAY ⟵ 1, MESMEN ⟵ 1, MAYBEB ⟵ PROD[1,2],
         MENBEB ⟵ PROD[1,2] e I ⟵ 2
13. Repetir con I desde 2 hasta 12
    13.1 Si MAYBEB < PROD[I,2]
                entonces
                        Hacer MAYBEB ⟵ PROD[I,2] y MESMAY ⟵ I
                sino
                13.1.1 Si MENBEB > PROD[I,2] entonces
```

www.FreeLibros.com

```
                    Hacer MENBEB ←—— PROD[I,2] y MESMEN ←—— I
            13.1.2 {Fin del condicional del paso 13.1.1}
    13.2 {Fin del condicional del paso 13.1}
         Hacer I ←—— I+1
14. {Fin del ciclo del paso 13}
15. Escribir "En el mes", MESMAY, "se registró el mayor costo de producción de
    bebidas, y en el mes", MESMEN, "el menor costo"
16. Hacer MENOR ←—— PROD[12,1], DEP ←—— 1 y J ←—— 2
17. Repetir con J desde 2 hasta 3
    17.1 Si MENOR > PROD[12,J] entonces
                    Hacer MENOR ←—— PROD[12,J] y DEP ←—— J
    17.2 {Fin del condicional del paso 17.1}
         Hacer J ←—— J+1
18. {Fin del ciclo del paso 17}
19. Escribir "El departamento ", DEP, "tuvo el menor costo en diciembre"
```

Problema 4.15

Una empresa automotriz necesita un programa para manejar los montos de ventas de sus N sucursales, a lo largo de los últimos M años. Los datos son dados de esta forma:

$$M, N$$
$$MONTO_{1,1}, MONTO_{1,2}, \ldots, MONTO_{1,N}$$
$$MONTO_{2,1}, MONTO_{2,2}, \ldots, MONTO_{2,N}$$
$$MONTO_{M,1}, MONTO_{M,2}, \ldots, MONTO_{M,N}$$

Donde:

M	es una variable de tipo entero que representa el número de años, $1 \leq M \leq 10$.
N	es una variable de tipo entero que expresa el número de sucursales de la empresa, $1 \leq N \leq 35$.
MONTO i,j	es una variable de tipo real que representa lo que se vendió en el año i en la sucursal j.

La información que necesitan los directores de la empresa para tomar decisiones es la siguiente:

a) Sucursal que más ha vendido en los M años.
b) Promedio de ventas por año.
c) Año con mayor promedio de ventas.

www.FreeLibros.com

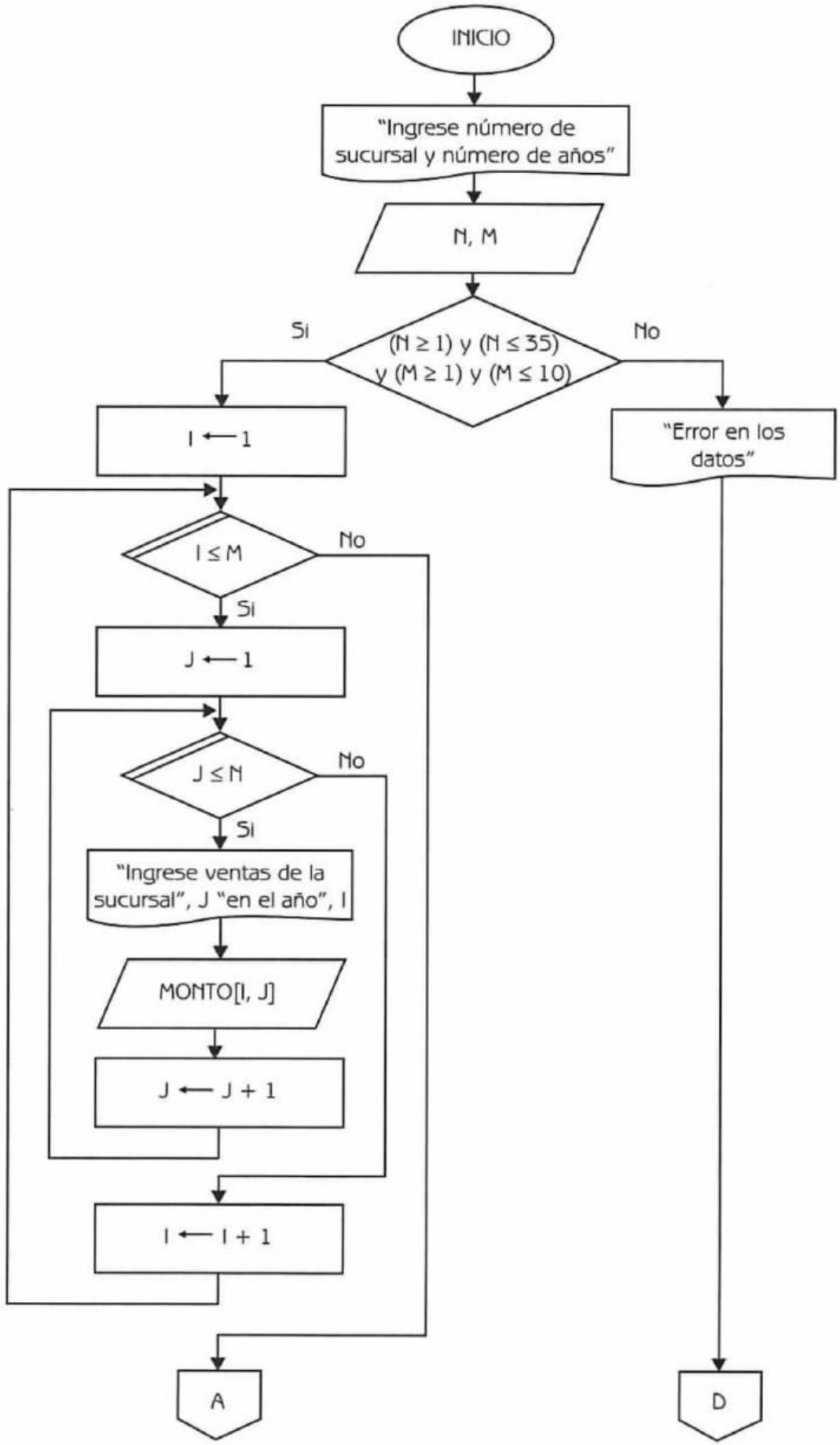
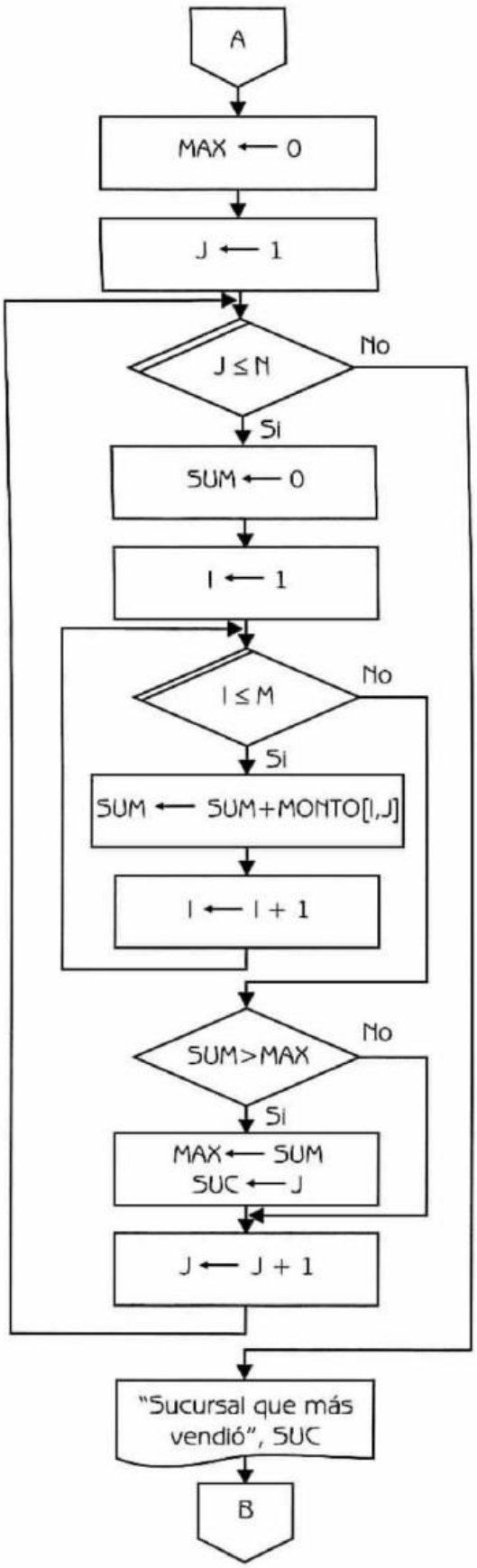


Diagrama de Flujo 4.23 (*continúa*)

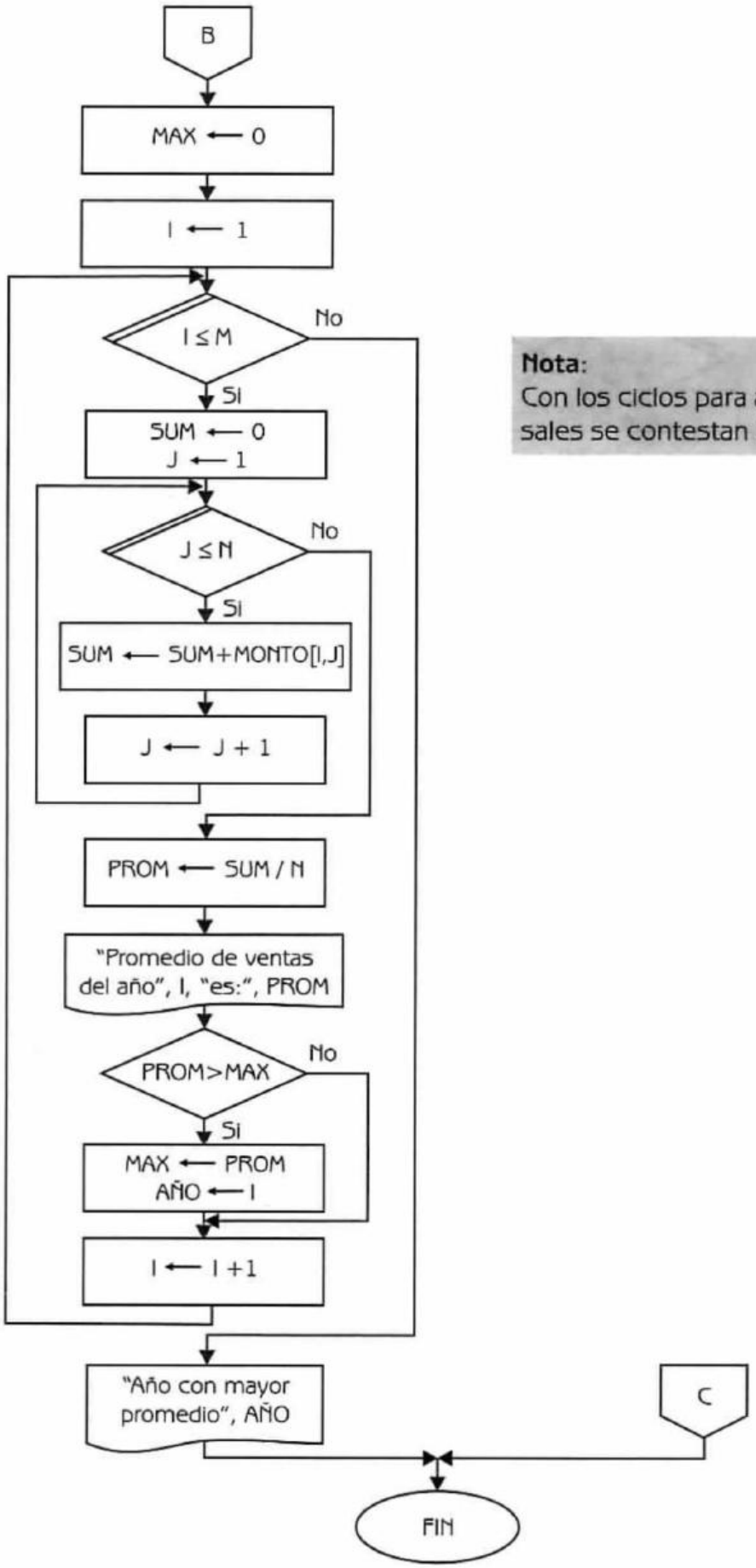
www.FreeLibros.com



Nota:
Cálculo de la sucursal que más ha vendido en los M años.

Diagrama de Flujo 4.23 (*continuación...*)

www.FreeLibros.com



Nota:
Con los ciclos para año y para sucursales se contestan los incisos b y c.

Diagrama de Flujo 4.23 (*continuación*)

www.FreeLibros.com

Explicación de variables

N, M: Variables de tipo entero.

MONTO: Arreglo bidimensional de tipo real.

I y *J*: Variables de tipo entero. Se utilizan como variables de control en varios ciclos.

MAX: Variable de tipo real. Se utiliza como auxiliar para determinar la sucursal que más vendió a lo largo de los M años, así como para calcular el año con mayor promedio de ventas.

SUM: Variable de tipo real. Se utiliza como acumulador para ventas por sucursal y por año.

SUC: Variable de tipo entero. Sirve para almacenar el número de sucursal que más vendió.

PROM: Variable de tipo real. Se usa para calcular el promedio anual de ventas.

AÑO: Variable de tipo entero. Sirve para almacenar el año con mayor promedio de ventas.

Programa 4.23

```
EMPRESA_AUTOMOTRIZ

{El programa, dada información sobre montos de ventas de N sucursales de una
empresa a lo largo de los últimos M años, obtiene lo siguiente: a) sucursal que más
ha vendido en los M años, b) promedio de ventas por año, y c) año con mayor pro-
medio de ventas}

{N, M, I, J, SUC y AÑO son variables de tipo entero. MAX, SUM y PROM son variables
de tipo real. MONTO es un arreglo bidimensional de tipo real}

1. Escribir "Ingrese número de sucursales y número de años"
2. Leer N y M
3. Si ((N >= 1) y (N <= 35)) y ((M >= 1) y (M <= 10))
      entonces
            Hacer I ←—— 1
      3.1 Repetir con I desde 1 hasta M
                  Hacer J ←—— 1
            3.1.1 Repetir con J desde 1 hasta N
```

www.FreeLibros.com

```
                    Escribir "Ingrese ventas de la sucursal", J,"en el año", I
                    Leer MONTO [I,J]
                    Hacer J ⟵ J+1
          3.1.2 {Fin del ciclo del paso 3.1.1}
                Hacer I ⟵ I+1
     3.2 {Fin del ciclo del paso 3.1}
          Hacer MAX ⟵ 0 y J ⟵ 1
     3.3 Repetir con J desde 1 hasta N
                Hacer SUM ⟵ 0 e I ⟵ 1
          3.3.1 Repetir con I desde 1 hasta M
                Hacer SUM ⟵ SUM + MONTO[I,J] e I ⟵ I+1
          3.3.2 {Fin del ciclo del paso 3.3.1}
          3.3.3 Si SUM > MAX entonces
                Hacer MAX ⟵ SUM y SUC ⟵ J
          3.3.4 {Fin del condicional del paso 3.3.3}
                Hacer J ⟵ J+1
     3.4 {Fin del ciclo del paso 3.3}
          Escribir "Sucursal que más vendió", SUC
          Hacer MAX ⟵ 0 e I ⟵ 1
     3.5 Repetir con I desde 1 hasta M
                Hacer SUM ⟵ 0 y J ⟵ 1
          3.5.1 Repetir con J desde 1 hasta N
                Hacer SUM ⟵ SUM+MONTO[I,J] y J ⟵ J+1
          3.5.2 {Fin del ciclo del paso 3.5.1}
                Hacer PROM ⟵ SUM/N
                Escribir "Promedio de ventas del año", I, "es :", PROM
          3.5.3 Si PROM > MAX entonces
                Hacer MAX ⟵ PROM y AÑO ⟵ I
          3.5.4 {Fin del condicional del paso 3.5.3}
                Hacer I ⟵ I+1
     3.6 {Fin del ciclo del paso 3.5}
          Escribir "Año con mayor promedio", AÑO
     sino
          Escribir "Error en los datos"
4. {Fin del condicional del paso 3}
```

www.FreeLibros.com

Problema 4.16

Sean A(M × N) y B(N) arreglos de dos y una dimensión, respectivamente. Construya un diagrama de flujo que asigne valores a B, a partir de A teniendo en cuenta los siguientes criterios:

a) $B[i] = \sum_{j=i}^{N} A[i, j]$ Si i es impar

b) $B[i] = \sum_{j=1}^{N} A[i, j] * A[i-1, j]$ Si i es par

Datos: A[1..M, 1..N], B[1..N] $1 \leq M \leq 30$ $1 \leq N \leq 20$

Donde:

A es un arreglo bidimensional de tipo real de M renglones y N columnas.

B es un arreglo unidimensional de tipo real de N columnas.

www.FreeLibros.com

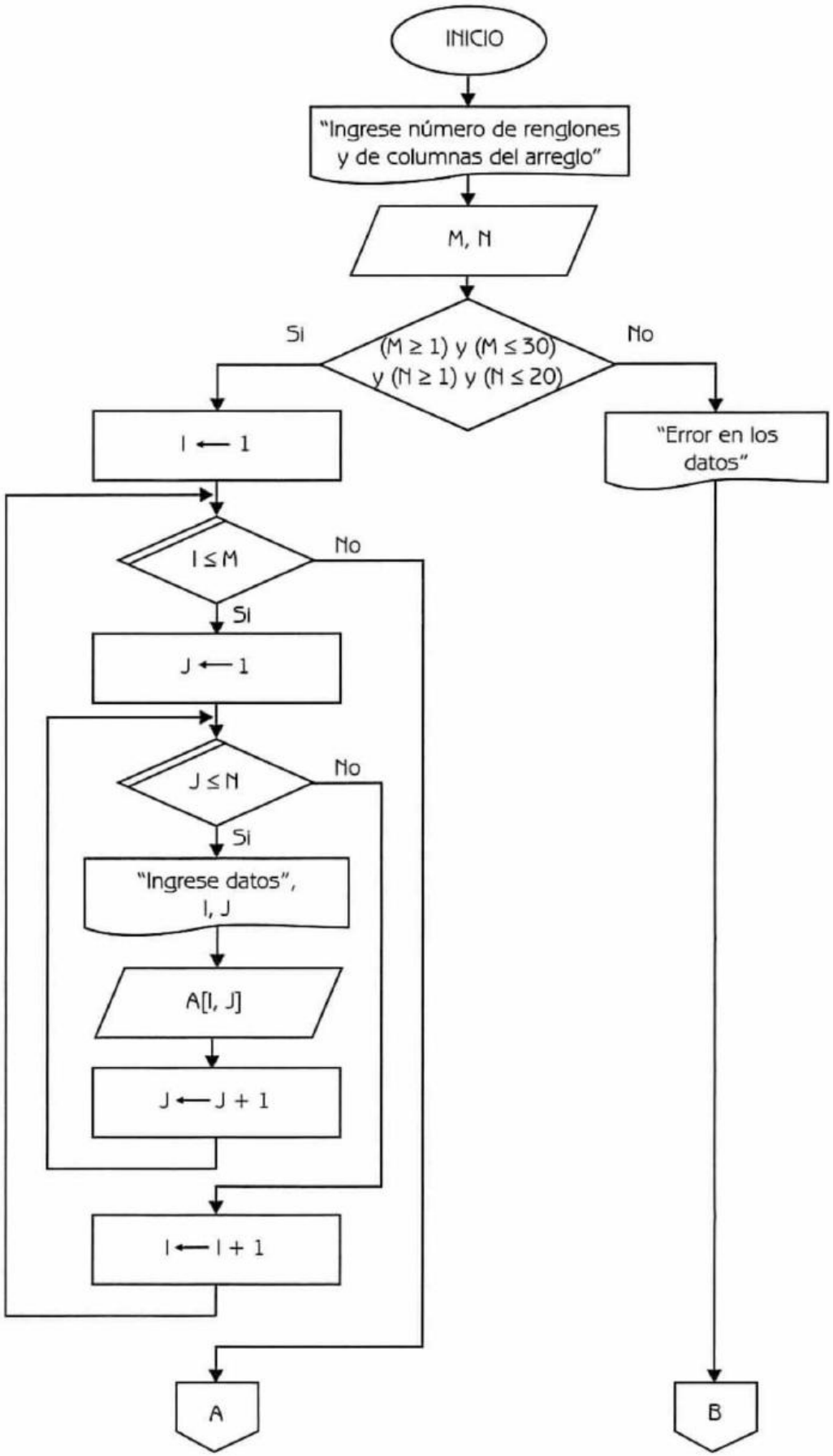


Diagrama de Flujo 4.24 (*continúa*)

www.FreeLibros.com

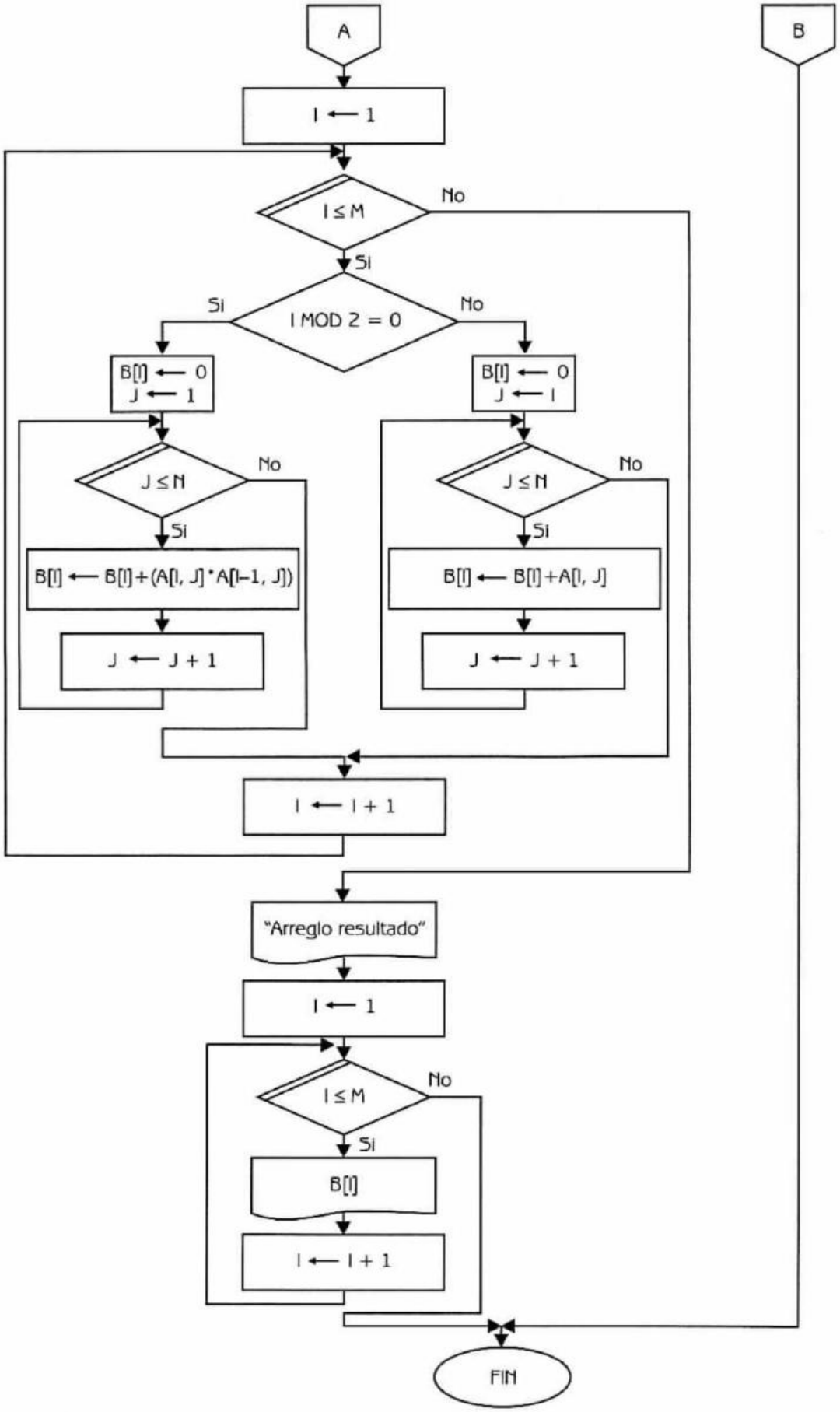


Diagrama de Flujo 4.24 (*continuación*)

www.FreeLibros.com

Explicación de las variables

M, N: Variables de tipo entero.
I, J: Variables de tipo entero. Se utilizan como variables de control de ciclos y como índices de los arreglos.
A: Arreglo bidimensional de tipo real.
B: Arreglo unidimensional de tipo real.

A continuación presentamos el diagrama de flujo en lenguaje algorítmico.

Programa 4.24

```
VECTOR_A_PARTIR_DE_MATRIZ_3

{El programa, dado un arreglo bidimensional A(M x N), asigna valores a un
arreglo unidimensional B, teniendo en cuenta ciertas reglas}

{M, N, I y J son variables de tipo entero. A es un arreglo bidimensional de tipo
real. B es un arreglo unidimensional de tipo real}

1. Escribir "Ingrese número de renglones y de columnas del arreglo"
2. Leer M y N
3. Si ((M >= 1) y (M <= 30)) y ((N >= 1) y (N <= 20))
      entonces
            Hacer I ⟵ 1
      3.1 Repetir con I desde 1 hasta M
                  Hacer J ⟵ 1
            3.1.1 Repetir con J desde 1 hasta N
                        Escribir "Ingrese dato", I, J
                        Leer A[I,J]
                        Hacer J ⟵ J + 1
            3.1.2 {Fin del ciclo del paso 3.1.1}
                  Hacer I ⟵ I + 1
      3.2 {Fin del ciclo del paso 3.1}
            Hacer I ⟵ 1
      3.3 Repetir con I desde 1 hasta M
            3.3.1. Si (I mod 2) = 0
                        entonces
                              Hacer B[I] ⟵ 0 y J ⟵ 1
                  3.3.1.1. Repetir con J desde 1 hasta N
                              Hacer B[I] ⟵ B[I] + (A[I,J] * A[I-1,J])
                                    y J ⟵ J + 1
```

www.FreeLibros.com

```
                3.3.1.2. {Fin del ciclo del paso 3.3.1.1}
                    sino
                        Hacer B[I] ⟵ 0 y J ⟵ 1
                3.3.1.3. Repetir con J desde 1 hasta N
                        Hacer B[I] ⟵ B[I] + A[I,J] y J ⟵ J + 1
                3.3.1.4. {Fin del ciclo del paso 3.3.1.3}
            3.3.2 {Fin del condicional del paso 3.3.1}
                Hacer I ⟵ I + 1
        3.4 {Fin del ciclo del paso 3.3}
            Escribir "Arreglo resultado"
            Hacer I ⟵ 1
        3.5 Repetir con I desde 1 hasta M
                Escribir B[I]
                Hacer I ⟵ I + 1
        3.6 {Fin del ciclo del paso 3.5}
        sino
            Escribir "Error en los datos"
    4. {Fin del condicional del paso 3}
```

Problema 4.17

Dada una matriz A(M × N) de tipo entero, construya un diagrama de flujo para calcular la traspuesta de dicha matriz. La traspuesta de una matriz se obtiene al escribir las filas de la matriz A como columnas. Por ejemplo si tenemos la siguiente matriz A:

$$A \begin{bmatrix} -5 & 6 & 8 & 4 \\ 2 & 5 & 7 & 9 \\ 1 & 3 & 4 & 6 \end{bmatrix}$$

La traspuesta de dicha matriz es la siguiente:

$$A^T \begin{bmatrix} -5 & 2 & 1 \\ 6 & 5 & 3 \\ 8 & 7 & 4 \\ 4 & 9 & 6 \end{bmatrix}$$

Dato: A[1..M, 1..N] $1 \leq M \leq 50$ $1 \leq N \leq 30$

Donde: A es un arreglo bidimensional de tipo entero.

www.FreeLibros.com

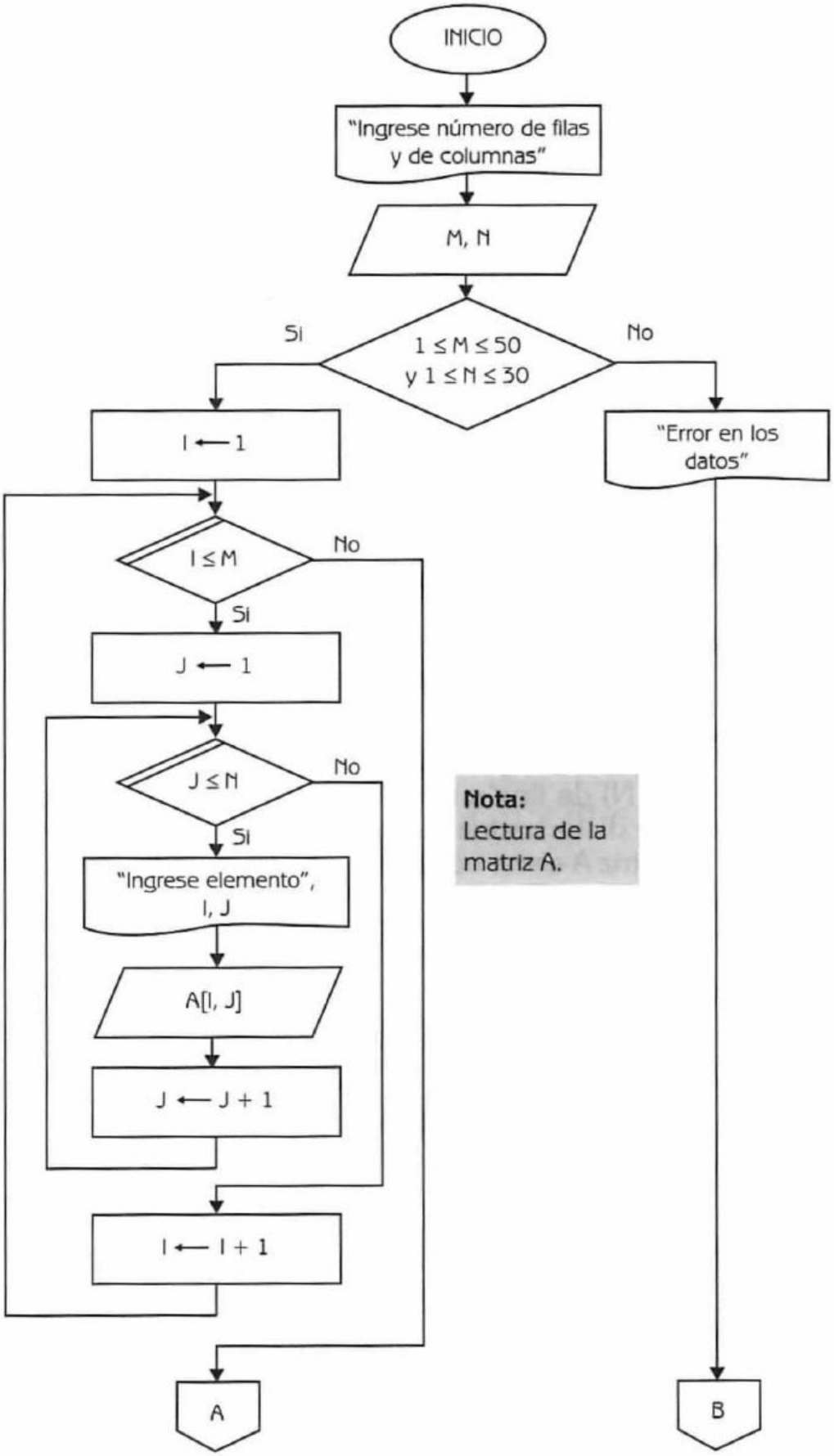


Diagrama de Flujo 4.25 (*continúa*)

www.FreeLibros.com

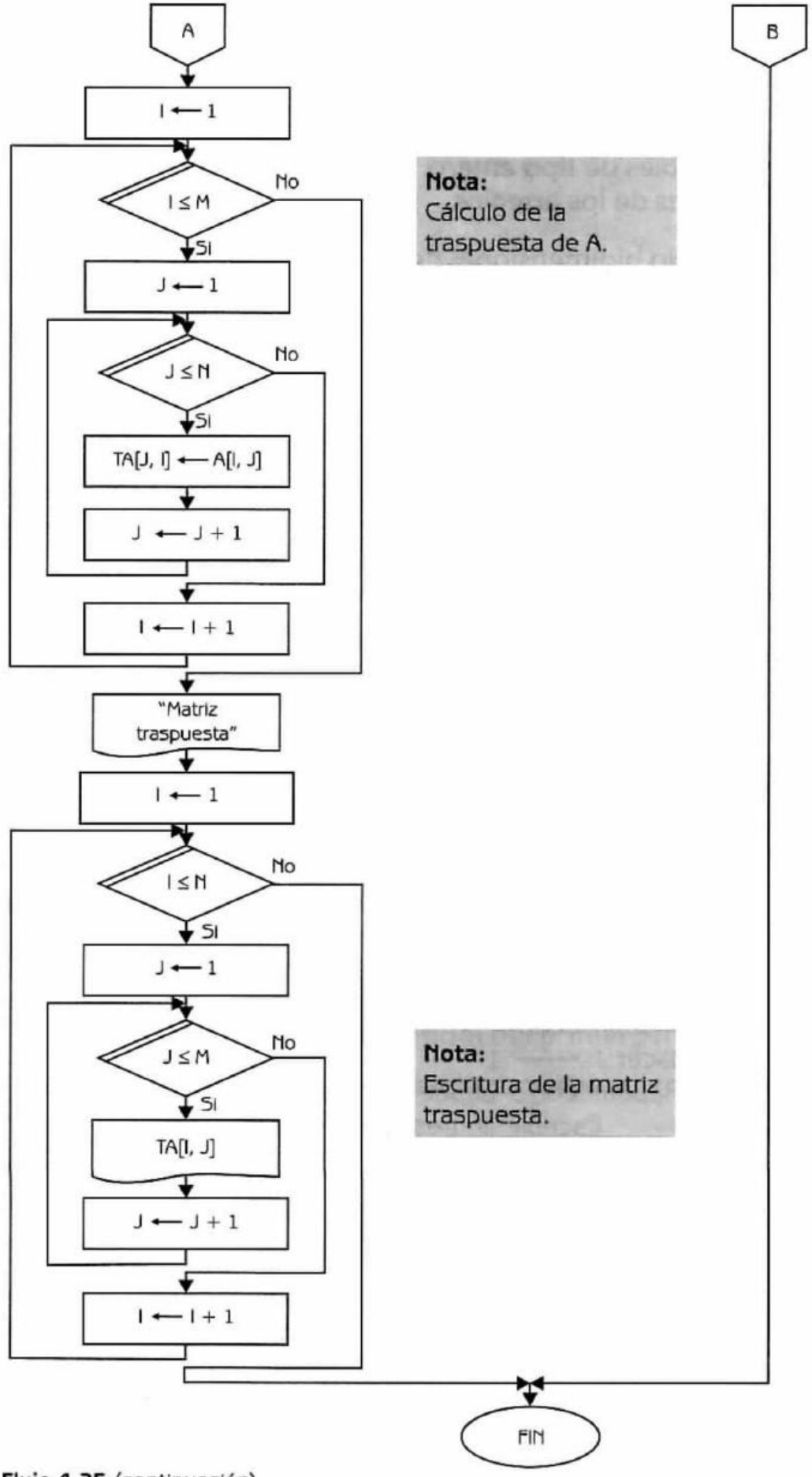


Diagrama de Flujo 4.25 (*continuación*)

www.FreeLibros.com

Explicación de las variables

M, N: Variables de tipo entero. Indican el número de renglones y de columnas.

I, J: Variables de tipo entero. Se usan para el control de ciclos y como índices de los arreglos.

A: Arreglo bidimensional de tipo entero.

TA: Arreglo bidimensional de tipo entero, TA[1..N, 1..M]. Almacena la traspuesta de la matriz dada.

A continuación presentamos el programa correspondiente.

Programa 4.25

```
TRASPUESTA_MATRIZ

{El programa dada una matriz A de M x N de tipo entero, calcula la traspuesta de dicha matriz}

{M, N, I y J son variables de tipo entero. A y TA son arreglos bidimensionales de tipo entero}

1. Escribir "Ingrese número de filas y de columnas"
2. Leer M y N
3. Si ((M >= 1 y M <= 50)) y ((N >= 1 y N <= 30))
        entonces
              Hacer I ←—— 1
        3.1 Repetir con I desde 1 hasta M
                    Hacer J ←—— 1
            3.1.1. Repetir con J desde 1 hasta N
                          Escribir "Ingrese elemento", I, J
                          Leer A[I,J]
                          Hacer J ←—— J + 1
            3.1.2 {Fin del ciclo del paso 3.1.1}
                    Hacer I ←—— I + 1
        3.2 {Fin del ciclo del paso 3.1}
              Hacer I ←—— 1
        3.3 Repetir con I desde 1 hasta M
                    Hacer J ←—— 1
            3.3.1 Repetir con J desde 1 hasta N
```

www.FreeLibros.com

```
                    Hacer TA[J,I] ⟵ A[I,J] y J ⟵ J + 1
        3.3.2 {Fin del ciclo del paso 3.3.1}
              Hacer I ⟵ I + 1
    3.4 {Fin del ciclo del paso 3.3}
        Escribir "Matriz traspuesta"
        Hacer I ⟵ 1
    3.5 Repetir con I desde 1 hasta N
              Hacer J ⟵ 1
        3.5.1 Repetir con J desde 1 hasta M
                    Escribir TA[I,J]
                    Hacer J ⟵ J + 1
        3.5.2 {Fin del ciclo del paso 3.5.1}
              Hacer I ⟵ I + 1
    3.6 {Fin del ciclo del paso 3.5}
     sino
        Escribir "Error en los datos"
4. {Fin del condicional del paso 3}
```

Problema 4.18

Escriba un diagrama de flujo que genere e imprima un cuadrado mágico de dimensión N (entero, positivo e impar). Un cuadrado mágico es una matriz cuadrada de orden N, que contiene los números naturales del uno al N * N, y donde la suma de cualquiera de los renglones, columnas o diagonales principales es siempre la misma. Puede utilizar los siguientes pasos para generar un cuadrado mágico:

a) El número 1 se coloca en la casilla central del primer renglón.

b) El siguiente número se coloca en la casilla correspondiente al renglón anterior y columna posterior.

c) El renglón anterior al primero es el último y la columna posterior a la última es la primera.

d) Si el número es un sucesor de un múltiplo de N, no se aplica la regla b. Se coloca en la casilla del renglón posterior y en la misma columna.

Dato: N (variable de tipo entero que expresa la dimensión del cuadrado mágico. N tiene que ser positivo e impar, $1 \leq N \leq 50$).

www.FreeLibros.com

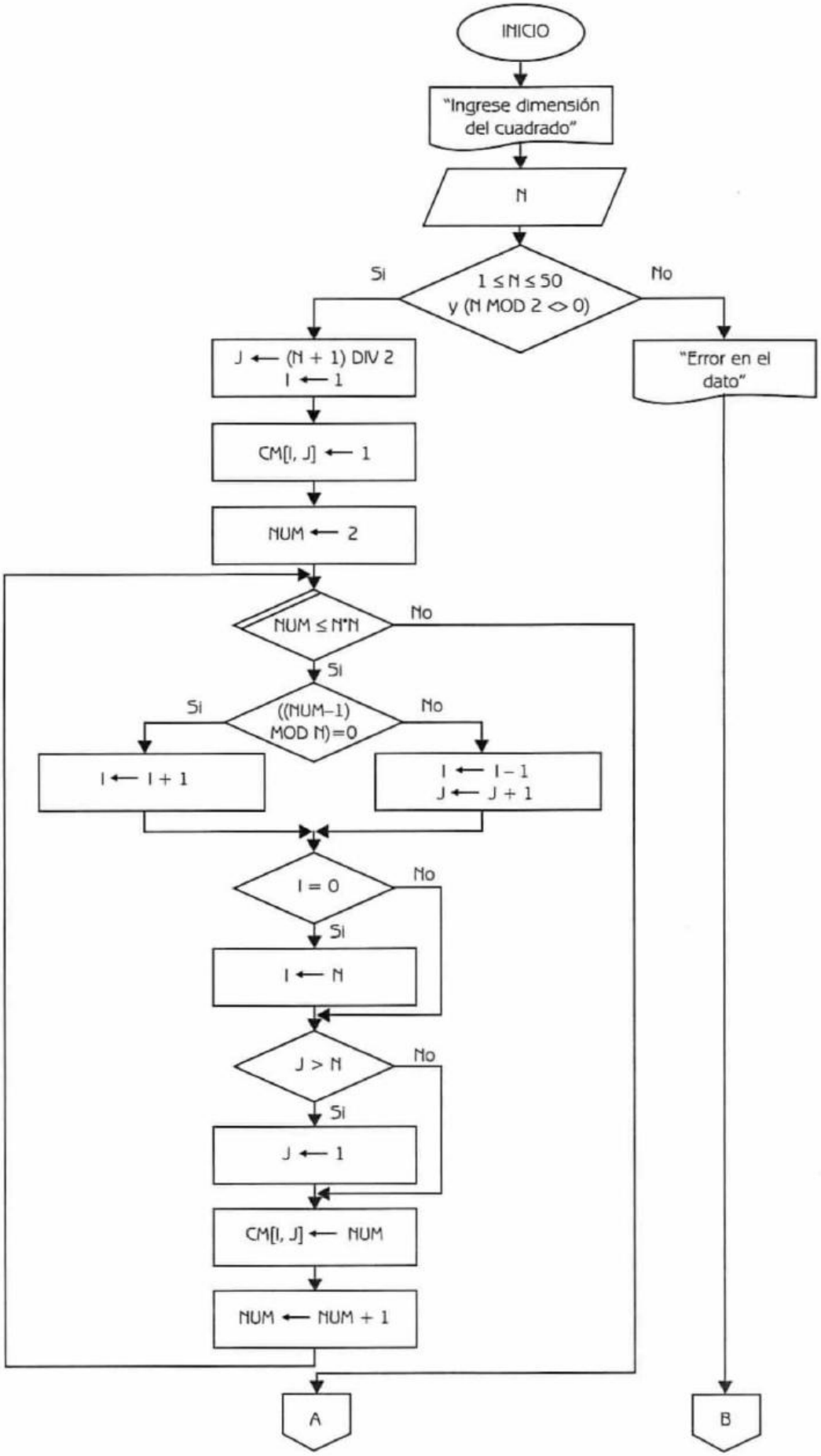


Diagrama de Flujo 4.26 (*continúa*)

www.FreeLibros.com

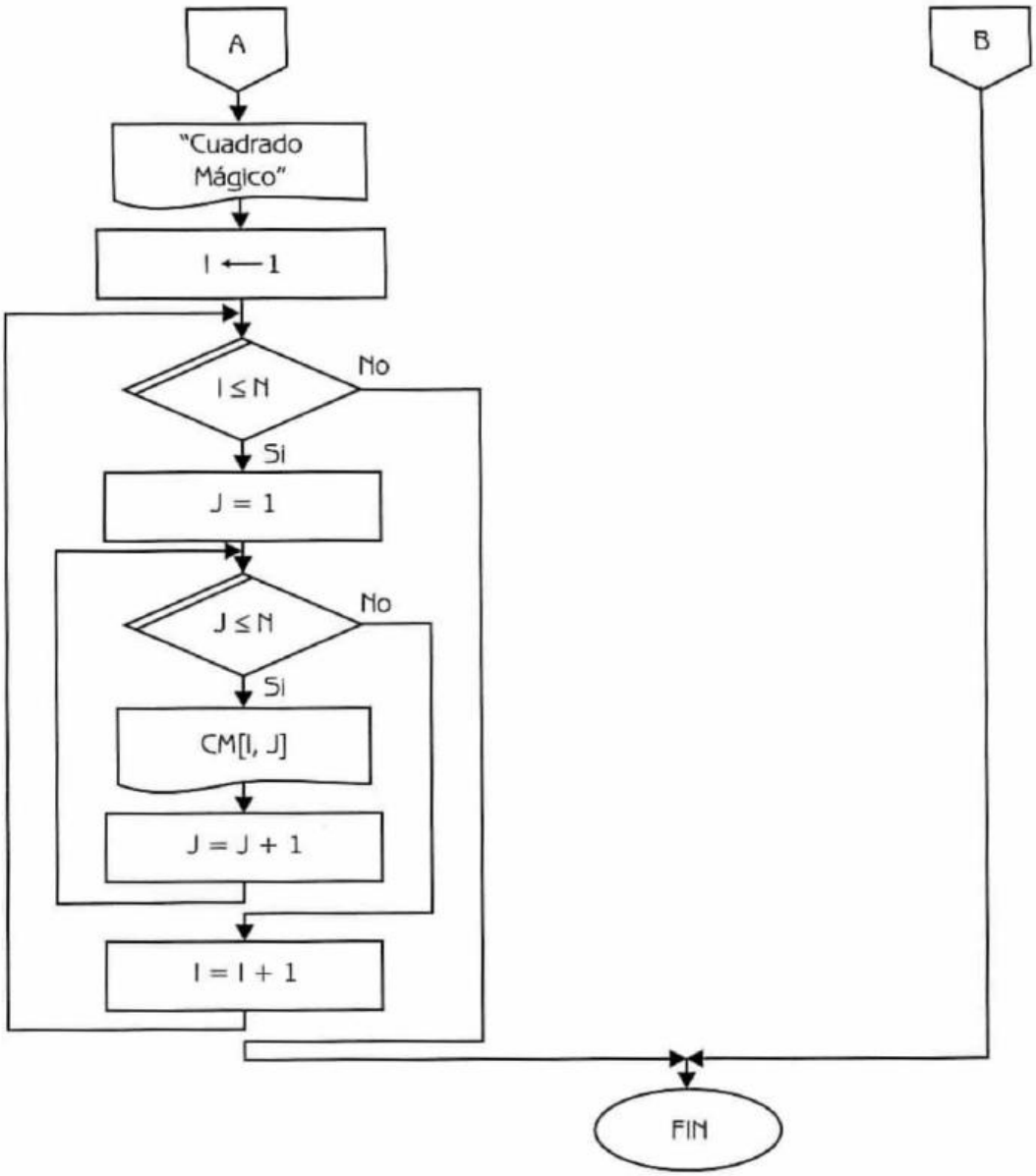


Diagrama de Flujo 4.26 (*continuación*)

Explicación de las variables

N: Variable de tipo entero.

I, J: Variables de tipo entero. Se utilizan como índices del arreglo.

CM: Arreglo bidimensional de tipo entero.

NUM: Variable de tipo entero. Se utiliza como variable de control del ciclo en el cual se forma el Cuadrado Mágico y sirve para almacenar los valores que se van asignando al Cuadrado.

A continuación presentamos el programa correspondiente.

www.FreeLibros.com

Programa 4.26

```
CUADRADO_MAGICO

{El programa genera e imprime un Cuadrado Mágico de dimensión N}

{I, N, J y NUM son variables de tipo entero. CM es un arreglo bidimensional de tipo
entero}

1. Escribir "Ingrese dimensión del cuadrado"
2. Leer N
3. Si ((N >= 1) y (N <= 50)) y (N mod 2 <> 0)
      entonces
            Hacer J ⟵ (N + 1) div 2, I ⟵ 1, CM[I,J] ⟵ 1 y NUM ⟵ 2
      3.1 Repetir con NUM desde 2 hasta N * N
         3.1.1. Si (((NUM - 1) mod N) = 0)
                     entonces
                           Hacer I ⟵ I + 1
                     sino
                           Hacer I ⟵ I - 1 y J ⟵ J + 1
         3.1.2 {Fin del condicional del paso 3.1.1}
         3.1.3 Si (I = 0) entonces
                     Hacer I ⟵ N
         3.1.4 {Fin del condicional del paso 3.1.3}
         3.1.5 Si (J > N) entonces
                     Hacer J ⟵ 1
         3.1.6 {Fin del condicional del paso 3.1.5}
               Hacer CM[I,J] ⟵ NUM y NUM ⟵ NUM + 1
      3.2 {Fin del ciclo del paso 3.1}
            Escribir "Cuadrado Mágico"
            Hacer I ⟵ 1
      3.3 Repetir con I desde 1 hasta N
               Hacer J ⟵ 1
         3.3.1 Repetir con J desde 1 hasta N
                     Escribir CM[I,J]
                     Hacer J ⟵ J + 1
         3.3.2 {Fin del ciclo del paso 3.3.1}
               Hacer I ⟵ I + 1
      3.4 {Fin del ciclo del paso 3.3}
       sino
            Escribir "Error en el dato"
4. {Fin del condicional del paso 3}
```

www.FreeLibros.com

Problema 4.19

En el arreglo tridimensional TEMP se almacenan los promedios de las temperaturas mensuales de los 32 estados de la República, durante los últimos 50 años.

Construya un diagrama de flujo que permita proporcionar la siguiente información.

a) El estado que tuvo en promedio la mayor temperatura durante los últimos diez años.

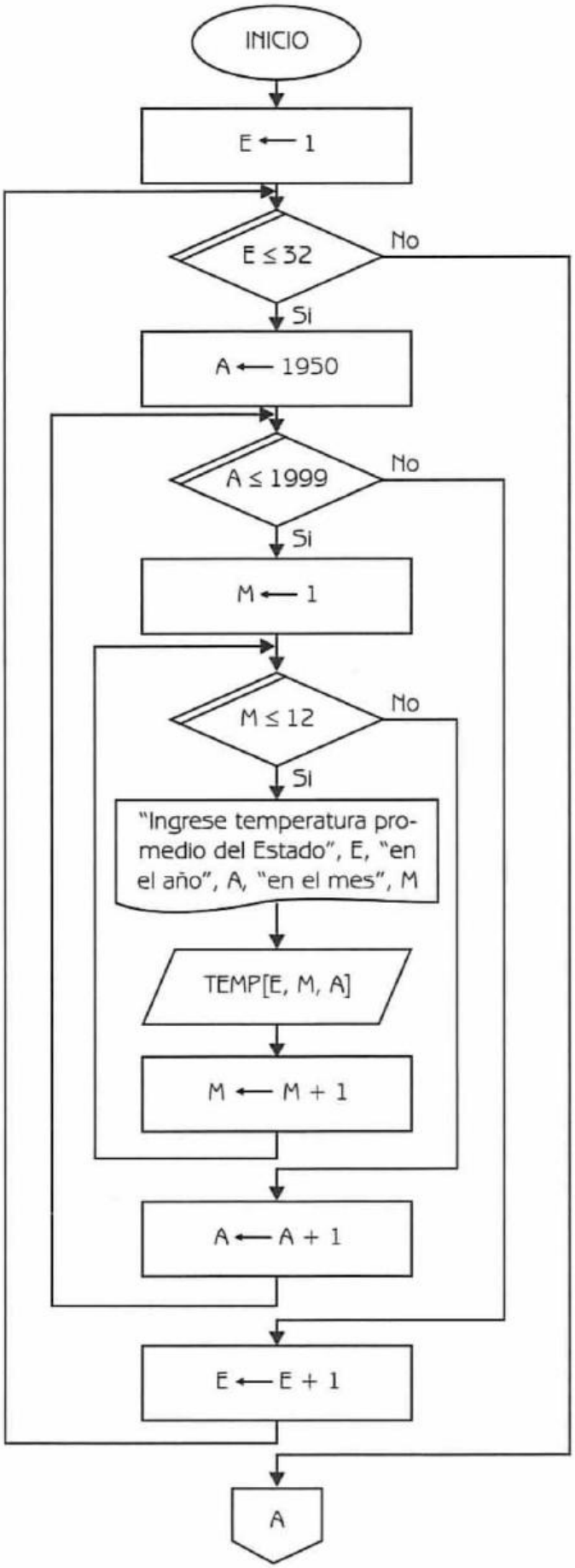
b) El estado con menor promedio anual de temperatura en el último año.

c) ¿En qué mes se registró el mayor promedio de temperaturas en el estado 29, en el año 1953?

d) ¿En qué mes y en qué estado se registró el menor promedio de temperaturas en 1975?

Dato: TEMP[1..32, 1..12, 1950..1999] (arreglo tridimensional de tipo real que almacena promedios mensuales de temperatura en los estados de la República, durante los últimos 50 años).

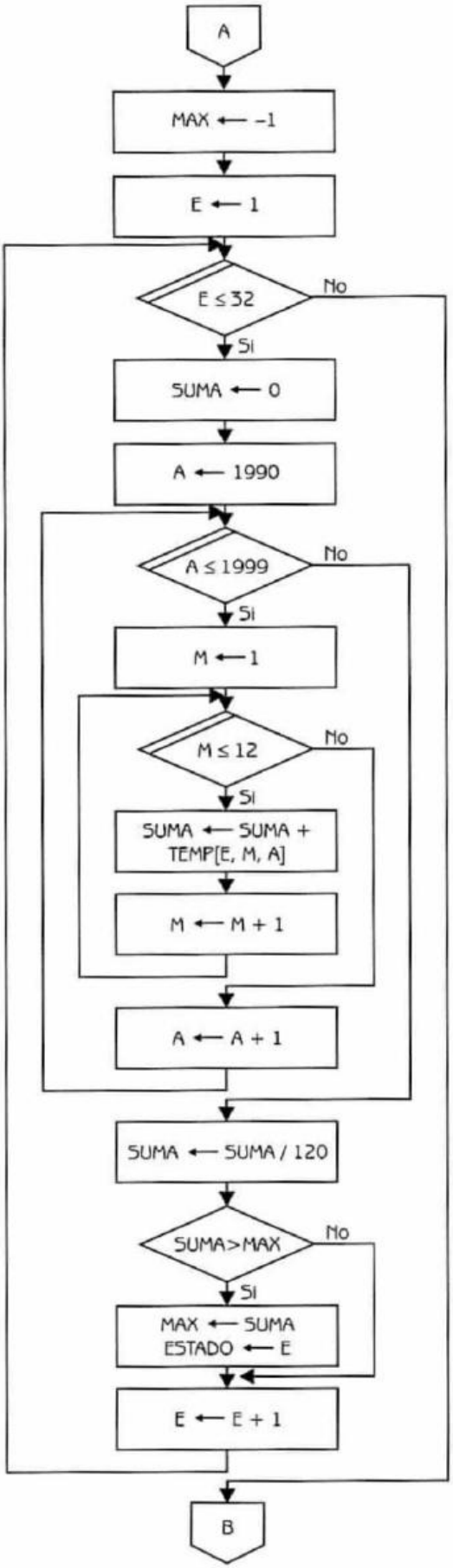
www.FreeLibros.com



Nota:
Lectura del arreglo tridimensional

Diagrama de Flujo 4.27 (*continúa*)

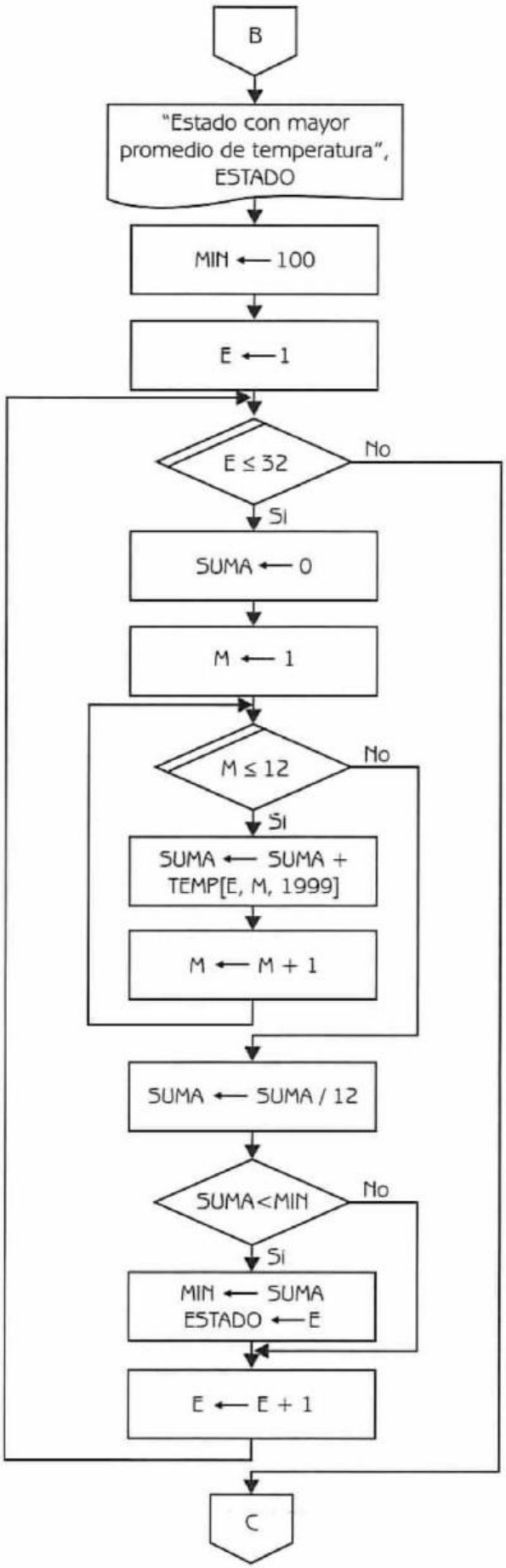
www.FreeLibros.com



Nota:
Cálculo del inciso a.

Diagrama de Flujo 4.27 (*continuación...*)

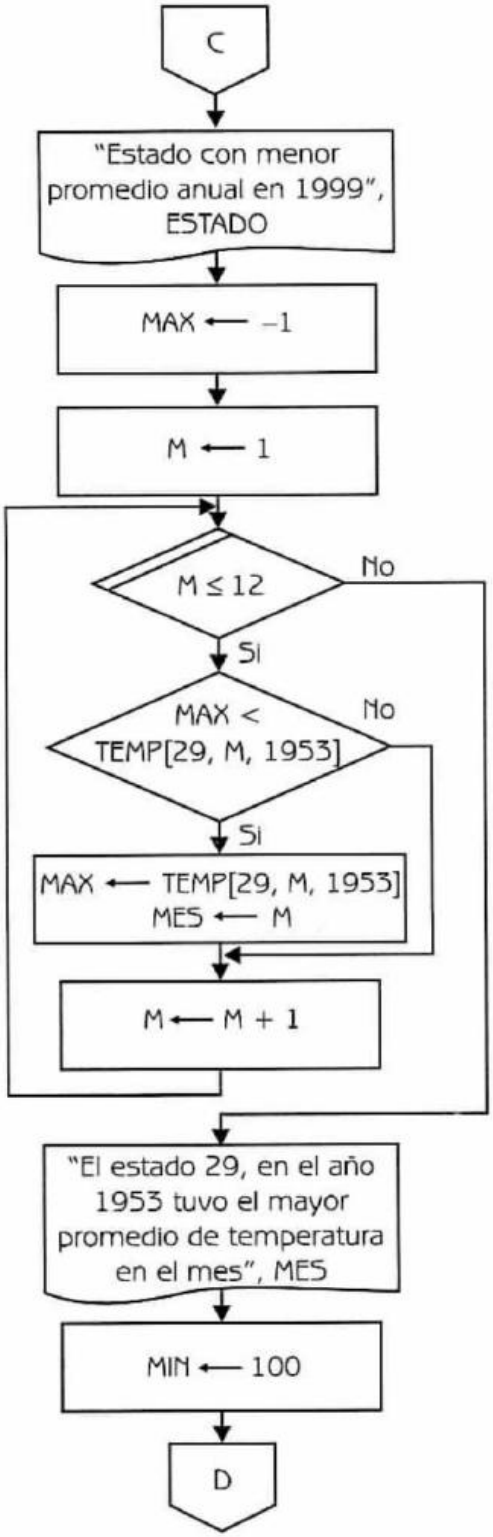
www.FreeLibros.com



Nota:
Cálculo del inciso b.

Diagrama de Flujo 4.27 (*continuación...*)

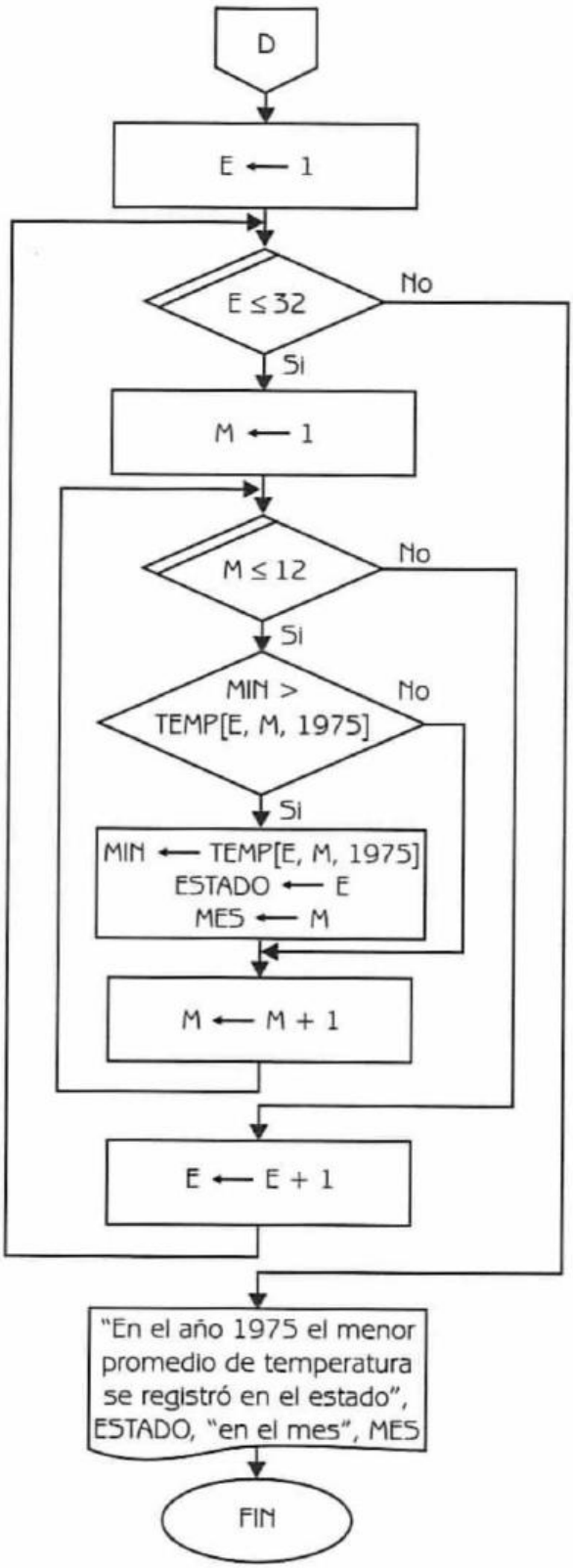
www.FreeLibros.com



Nota:
Cálculo del inciso c.

Diagrama de flujo 4.27 (*continuación...*)

www.FreeLibros.com



Nota:
Cálculo del inciso d.

Diagrama de flujo 4.27 (*continuación*)

www.FreeLibros.com

Explicación de las variables

E, M, A: Variables de tipo entero. Se utilizan como índice del arreglo y como variables de control de diversos ciclos.

TEMP: Arreglo tridimensional de tipo real.

MAX: Variable de tipo real. Se utiliza como variable auxiliar en la búsqueda de máximas temperaturas promedio.

SUMA: Variable de tipo real. Se utiliza para acumular temperaturas y para el cálculo de promedios.

ESTADO: Variable de tipo entero. Almacena el número del estado con mayor o menor temperatura promedio, según lo solicitado en los incisos a, b y d.

MIN: Variable de tipo real. Se usa como variable auxiliar en la búsqueda del menor promedio de temperaturas, incisos b y d.

MES: Variable de tipo entero. Almacena el número de mes en el cual se ha registrado cierta temperatura, incisos c y d.

Programa 4.27

```
TEMPERATURAS

{El programa obtiene información estadística relevante, respecto a información al-
macenada en forma mensual sobre promedios de temperaturas en los 32 estados
de la República, en los últimos 50 años}

{E, M, A, ESTADO y MES son variables de tipo entero. MAX, SUMA y MIN son variables
de tipo real. TEMP es un arreglo tridimensional de tipo real}

1. Hacer E ←— 1
2. Repetir con E desde 1 hasta 32
          Hacer A ←— 1950
      2.1 Repetir A desde 1950 hasta 1999
                Hacer M ←— 1
          2.1.1 Repetir M desde 1 hasta 12
                      Escribir "Ingrese temperatura promedio del estado", E,
                               "en el año", A, "en el mes", M
                      Leer TEMP[E,M,A]
                      Hacer M ←— M+1
          2.1.2 {Fin del ciclo del paso 2.1.1}
                Hacer A ←— A+1
```

www.FreeLibros.com

```
        2.2 {Fin del ciclo del paso 2.1}
            Hacer E ⟵ E+1
3. {Fin del ciclo del paso 2}
4. Hacer MAX ⟵ -1 y E ⟵ 1
5. Repetir con E desde 1 hasta 32
            Hacer SUMA ⟵ 0 y A ⟵ 1990
        5.1 Repetir con A desde 1990 hasta 1999
                Hacer M ⟵ 1
            5.1.1 Repetir con M desde 1 hasta 12
                    Hacer SUMA ⟵ SUMA+TEMP[E,M,A] y M ⟵ M+1
            5.1.2 {Fin del ciclo del paso 5.1.1}
                Hacer A ⟵ A+1
        5.2 {Fin del ciclo del paso 5.1}
            Hacer SUMA ⟵ SUMA/120
        5.3 Si SUMA > MAX entonces
                Hacer MAX ⟵ SUMA y ESTADO ⟵ E
        5.4 {Fin del condicional del paso 5.3}
            Hacer E ⟵ E+1
6. {Fin del ciclo del paso 5}
7. Escribir "Estado con mayor promedio de temperatura", ESTADO
8. Hacer MIN ⟵ 100 y E ⟵ 1
9. Repetir con E desde 1 hasta 32
            Hacer SUMA ⟵ 0 y M ⟵ 1
        9.1 Repetir con M desde 1 hasta 12
                Hacer SUMA ⟵ SUMA + TEMP[E,M,1999] y M ⟵ M+1
        9.2 {Fin del ciclo del paso 9.1}
            Hacer SUMA ⟵ SUMA/12
        9.3 Si SUMA < MIN entonces
                Hacer MIN ⟵ SUMA y ESTADO ⟵ E
        9.4 {Fin del condicional del paso 9.3}
            Hacer E ⟵ E+1
10. {Fin del ciclo del paso 4}
11. Escribir "Estado con menor promedio anual en 1999", ESTADO
12. Hacer MAX ⟵ -1 y M ⟵ 1
13. Repetir con M desde 1 hasta 12
        13.1 Si MAX < TEMP[29,M,1953] entonces
                Hacer MAX ⟵ TEMP[29,M,1953] y MES ⟵ M
        13.2 {Fin del condicional del paso 13.1}
            Hacer M ⟵ M+1
14. {Fin del ciclo del paso 13}
15. Escribir "El estado 29, en el año 1953 tuvo el mayor promedio de
            temperatura en el mes", MES
```

www.FreeLibros.com

```
16. Hacer MIN ←—— 100 y E ←—— 1
17. Repetir con E desde 1 hasta 32
          Hacer M ←—— 1
      17.1 Repetir con M desde 1 hasta 12
          17.1.1 Si MIN > TEMP[E,M,1975] entonces
                        Hacer MIN ←—— TEMP[E,M,1975], ESTADO ←—— E y
                                  MES ←—— M
          17.1.2 {Fin del condicional del paso 17.1.1}
                    Hacer M ←—— M+1
      17.2 {Fin del ciclo del paso 17}
           Hacer E ←—— E+1
18. {Fin del ciclo del paso 17}
19. Escribir "En el año 1975 el menor promedio de temperatura se registró en
          el estado", ESTADO, "en el mes", MES
```

Problema 4.20

En el INAH se almacena información sobre el número de visitantes a los diferentes centros arqueológicos (58), en forma mensual en los últimos 10 años. La información se registra en un arreglo tridimensional.

Construya un diagrama de flujo que pueda proporcionar la siguiente información.

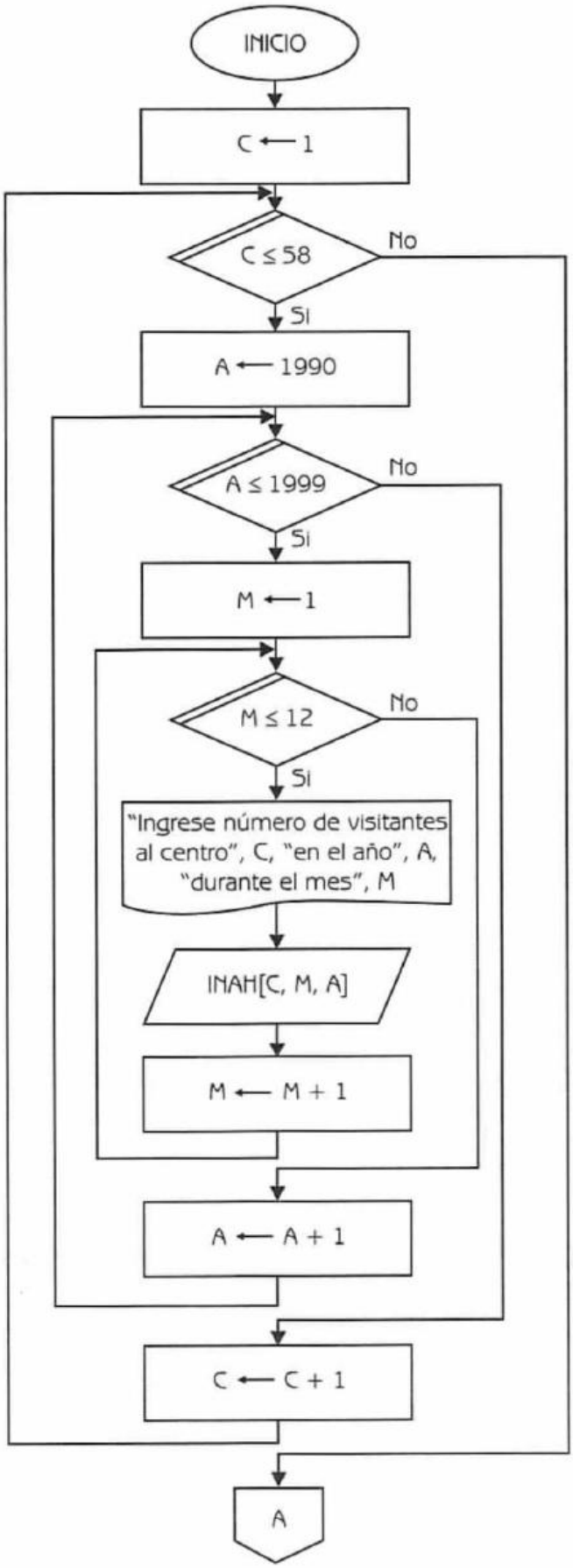
a) ¿En qué año, el centro arqueológico 43, obtuvo la mayor afluencia de visitantes?

b) ¿Cuál fue el centro arqueológico que en 1994 recibió la mayor afluencia de visitantes?

c) ¿En qué mes de 1995 los centros recibieron mayor número de visitantes?

Dato: INAH [1..58, 1..12, 1990..1999] (arreglo tridimensional de tipo entero que registra la afluencia de visitantes a los diferentes centros arqueológicos del país, en los últimos 10 años).

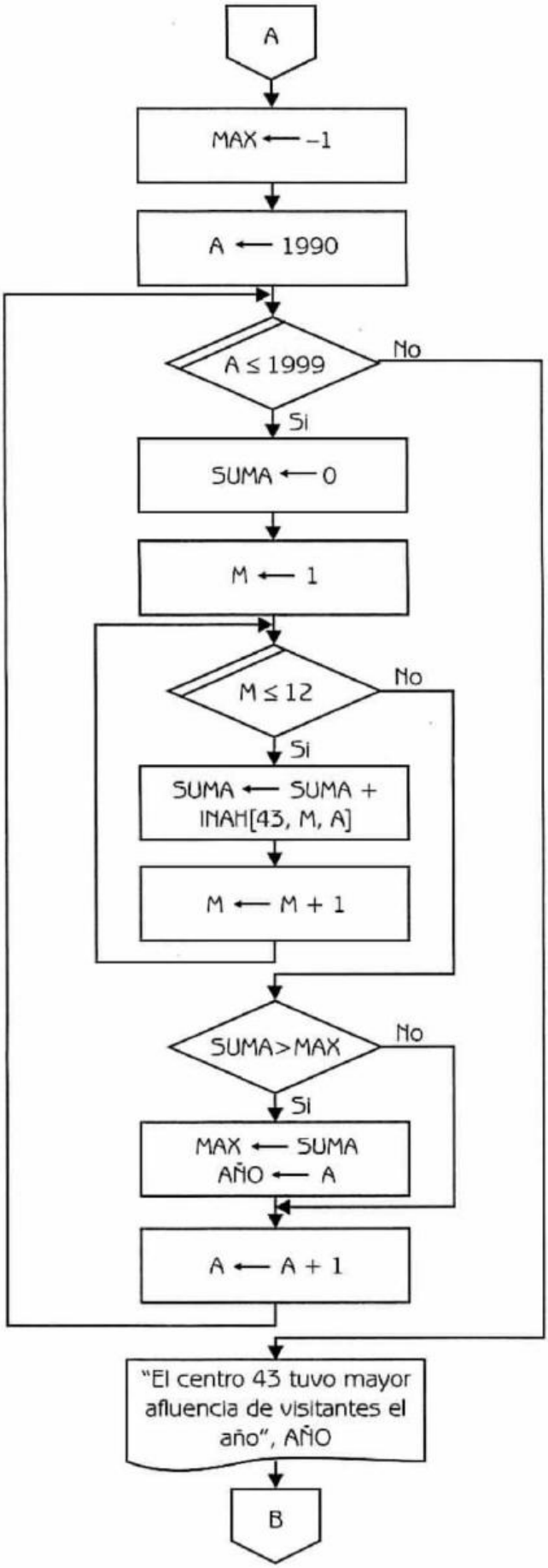
www.FreeLibros.com



Nota:
Lectura del arreglo tridimensional.

Diagrama de Flujo 4.28 (*continúa*)

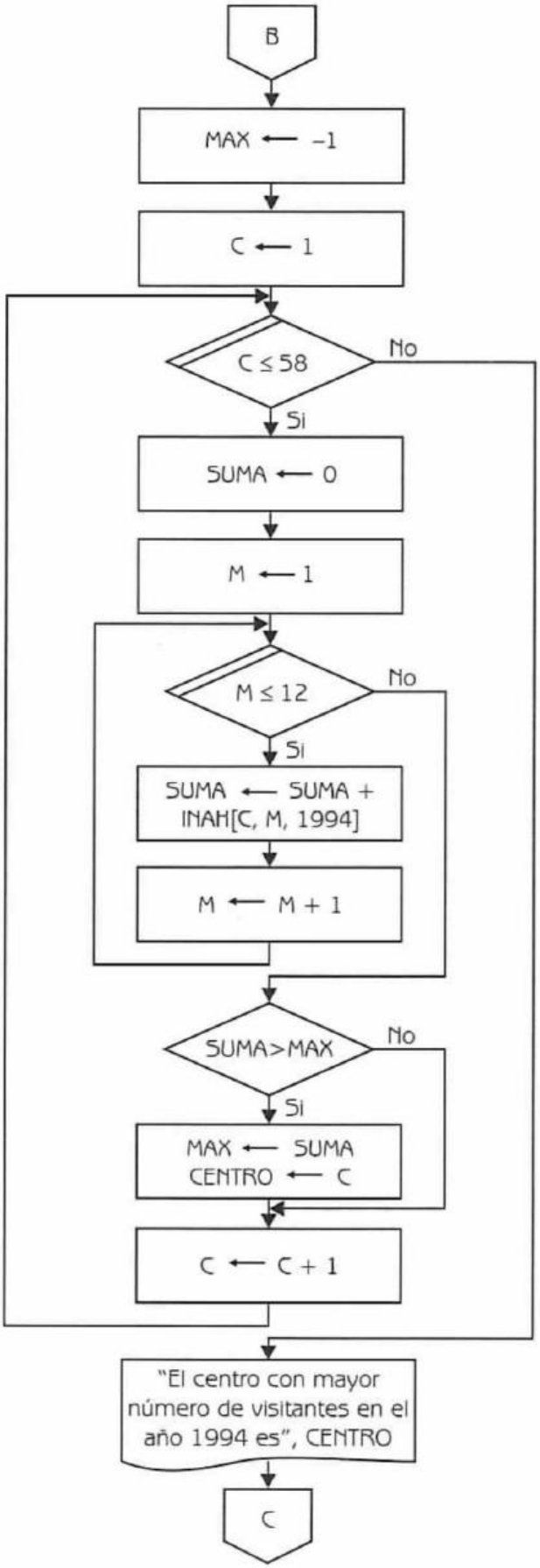
www.FreeLibros.com



Nota:
Cálculo del inciso a.

Diagrama de flujo 4.28 (*continuación...*)

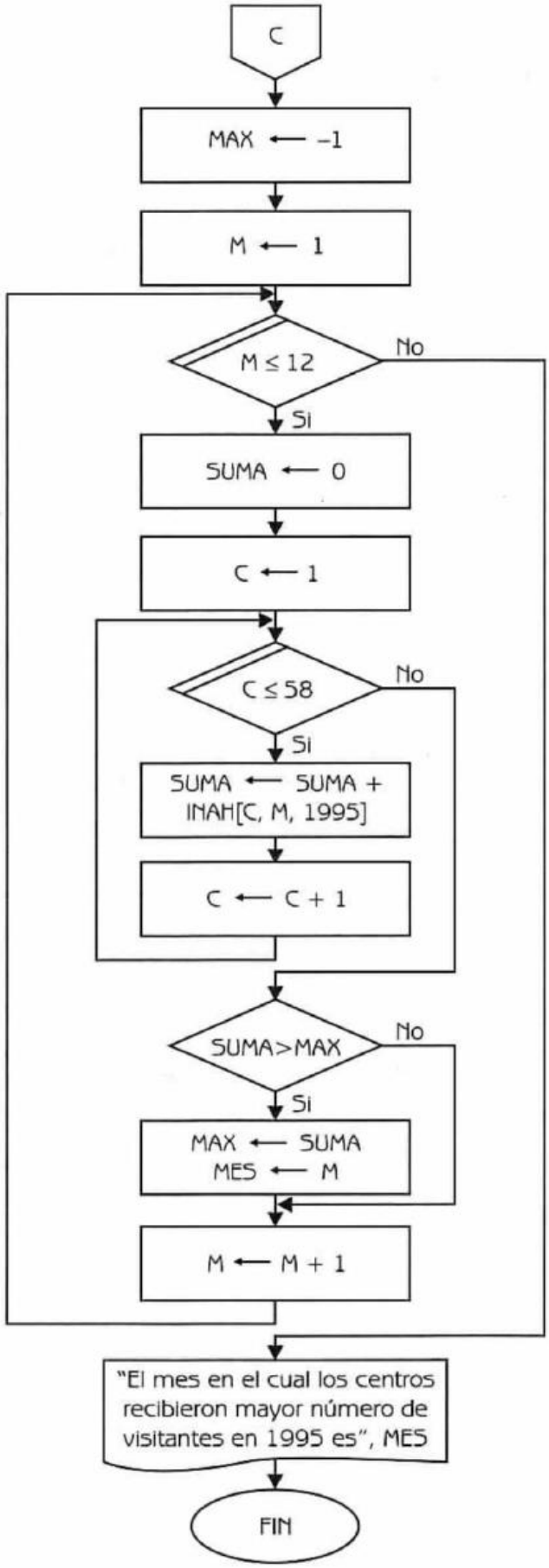
www.FreeLibros.com



Nota:
Cálculo del inciso b.

Diagrama de flujo 4.28 (*continuación...*)

www.FreeLibros.com



Nota:
Cálculo del inciso c.

Diagrama de flujo 4.28 (*continuación*)

www.FreeLibros.com

Explicación de las variables

C, M y A: Variables de tipo entero. Se utilizan como índice del arreglo y como variables de control de los ciclos.

INAH: Arreglo tridimensional de tipo entero.

MAX: Variable de tipo entero. Se utiliza para almacenar los totales de visitantes por año, centro y mes, según lo pide cada inciso del problema.

SUMA: Variable de tipo entero. Se utiliza para sumar el número de visitantes por año, centro y mes, de acuerdo a lo solicitado en cada inciso.

AÑO: Variable de tipo entero. Almacena el año con mayor número de visitantes en el centro 43.

CENTRO: Variable de tipo entero. Almacena el centro con mayor número de visitantes en el año 1994.

MES: Variable de tipo entero. Almacena el mes con mayor número de visitantes en el año 1985, considerando todos los centros.

A continuación presentamos al diagrama de flujo en lenguaje algorítmico.

Programa 4.28

```
INAH

{El programa obtiene información estadística muy valiosa para el INAH}

{C, M, A, MAX, SUMA, AÑO, CENTRO y MES son variables de tipo entero. INAH
es un arreglo tridimensional de tipo entero}

1. Hacer C ←— 1
2. Repetir con C desde 1 hasta 58
        Hacer A ←— 1990
   2.1 Repetir con A desde 1990 hasta 1999
            Hacer M ←— 1
     2.1.1 Repetir con M desde 1 hasta 12
               Escribir "Ingrese número de visitantes al centro", C, "en el año", A,
                        "durante el mes", M
               Leer INAH[C,M,A]
               Hacer M ←— M + 1
     2.1.2 {Fin del ciclo del paso 2.1.1}
```

www.FreeLibros.com

```
            Hacer A ⟵ A + 1
    2.2 {Fin del ciclo del paso 2.1}
        Hacer C ⟵ C + 1
3. {Fin del ciclo del paso 2}
4. Hacer MAX ⟵ -1 y A ⟵ 1990
5. Repetir con A desde 1990 hasta 1999
        Hacer SUMA ⟵ 0 y M ⟵ 1
    5.1 Repetir con M desde 1 hasta 12
            Hacer SUMA ⟵ SUMA + INAH[43,M,A] y M ⟵ M + 1
    5.2 {Fin del ciclo del paso 5.1}
    5.3 Si SUMA > MAX entonces
            Hacer MAX ⟵ SUMA y AÑO ⟵ A
    5.4 {Fin del condicional del paso 5.3}
        Hacer A ⟵ A + 1
6. {Fin del ciclo del paso 5}
7. Escribir "El centro 43 tuvo mayor afluencia de visitantes el año", AÑO
8. Hacer MAX ⟵ -1 y C ⟵ 1
9. Repetir con C desde 1 hasta 58
        Hacer SUMA ⟵ 0 y M ⟵ 1
    9.1 Repetir con M desde 1 hasta 12
            Hacer SUMA ⟵ SUMA + INAH[C,M,1994] y M ⟵ M + 1
    9.2 {Fin del ciclo del paso 9.1}
    9.3 Si SUMA > MAX entonces
            Hacer MAX ⟵ SUMA y CENTRO ⟵ C
    9.4 {Fin del condicional del paso 9.3}
        Hacer C ⟵ C + 1
10. {Fin del ciclo del paso 9}
11. Escribir "El centro con mayor número de visitantes en el año 1994 es",
        CENTRO
12. Hacer MAX ⟵ -1 y M ⟵ 1
13. Repetir con M desde 1 hasta 12
        Hacer SUMA ⟵ 0 y C ⟵ 1
    13.1 Repetir con C desde 1 hasta 58
            Hacer SUMA ⟵ SUMA + INAH[C,M,1995] y C ⟵ C + 1
    13.2 {Fin del ciclo del paso 13.1}
    13.3 Si SUMA > MAX entonces
            Hacer MAX ⟵ SUMA y MES ⟵ M
    13.4 {Fin del condicional del paso 13.3}
        Hacer M ⟵ M + 1
14. {Fin del ciclo del paso 13}
15. Escribir "El mes en el cual los centros recibieron mayor número de visitantes
        en 1995 es", MES
```

www.FreeLibros.com

Problema 4.21

En dos arreglos unidimensionales (MAT y CAL) se almacenan las matrículas y promedio de calificaciones de los alumnos de una preparatoria. Construya un diagrama de Flujo que permita obtener la siguiente información:

a) La matrícula del alumno con el mayor promedio.

b) La matrícula del alumno con el menor promedio.

c) El promedio general de los alumnos.

Datos: MAT [1..N], CAL [1..N] $1 \leq N \leq 500$

Donde: MAT es un arreglo unidimensional de tipo entero que almacena las matrículas de los alumnos.
CAL es un arreglo unidimensional de tipo real que almacena los promedios de los alumnos.

www.FreeLibros.com

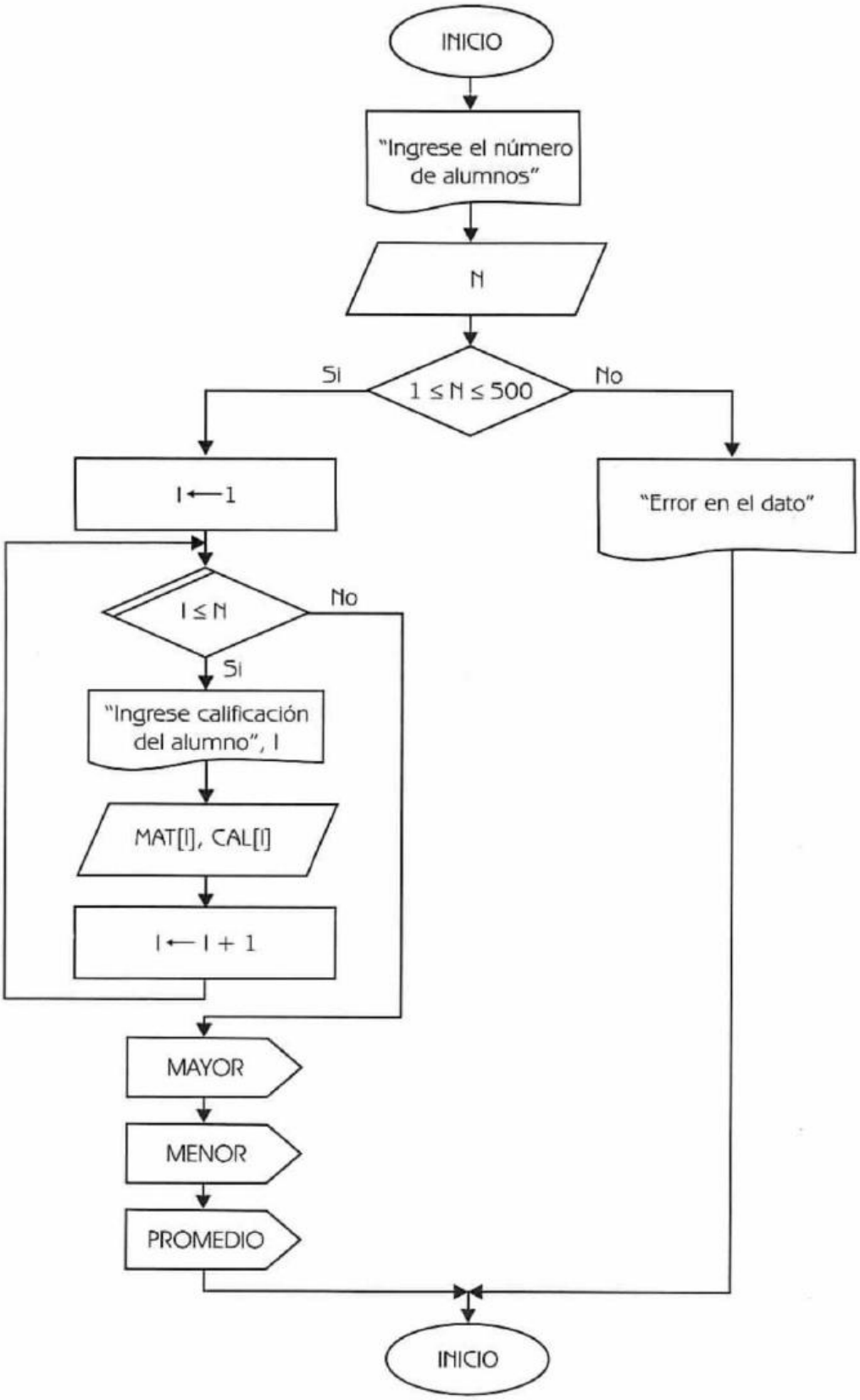


Diagrama de flujo 4.29 (*continúa*)

www.FreeLibros.com

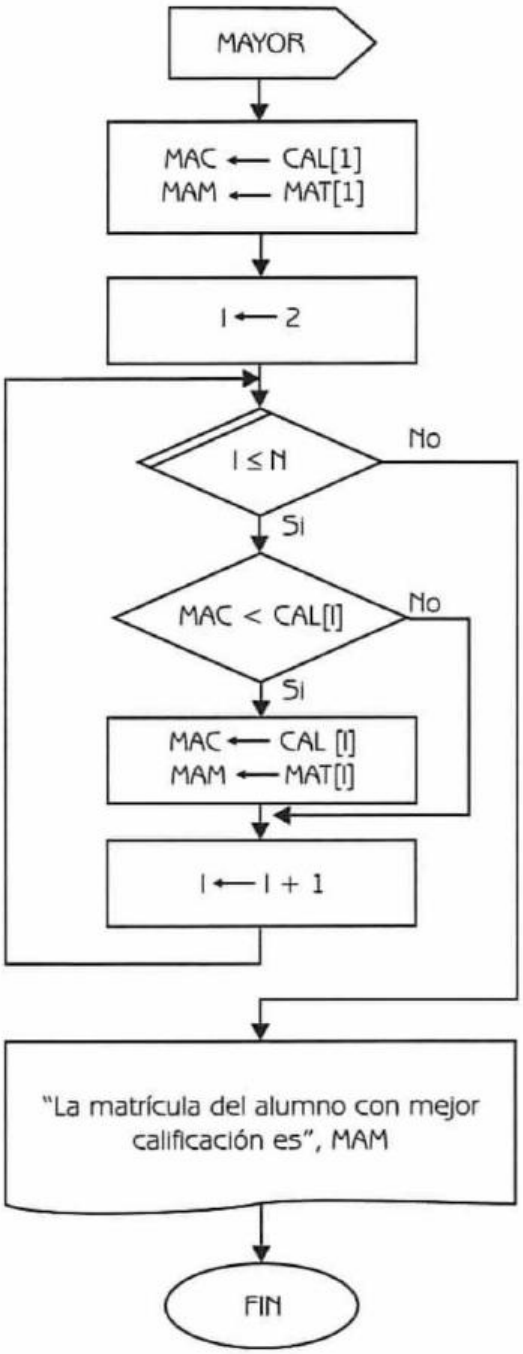


Diagrama de flujo 4.29 (*continuación...*)

www.FreeLibros.com

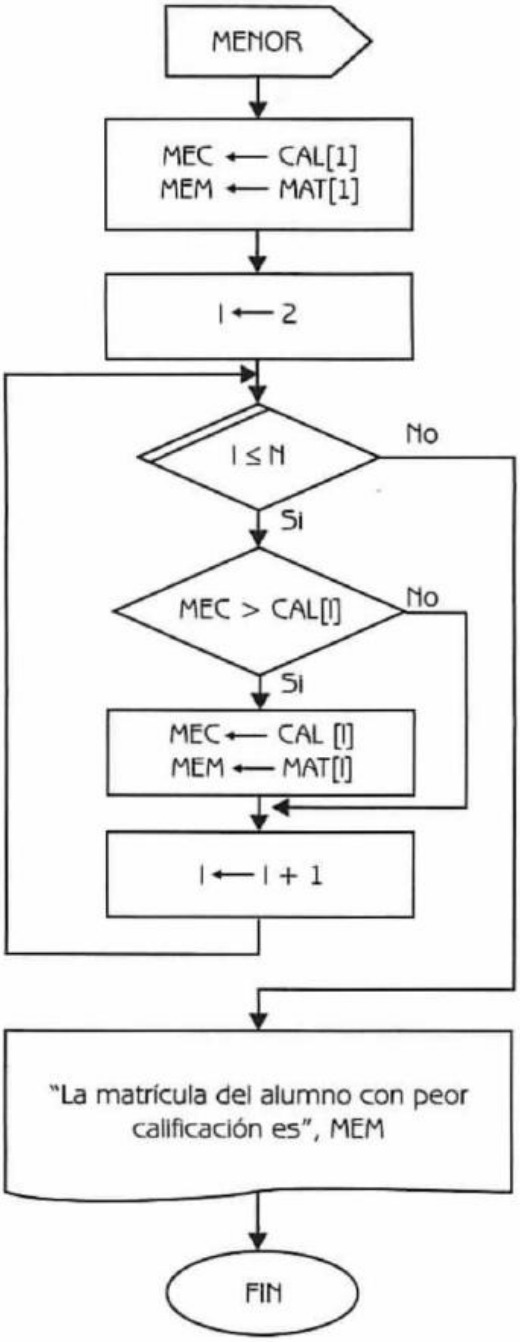


Diagrama de flujo 4.29 (*continuación...*)

www.FreeLibros.com

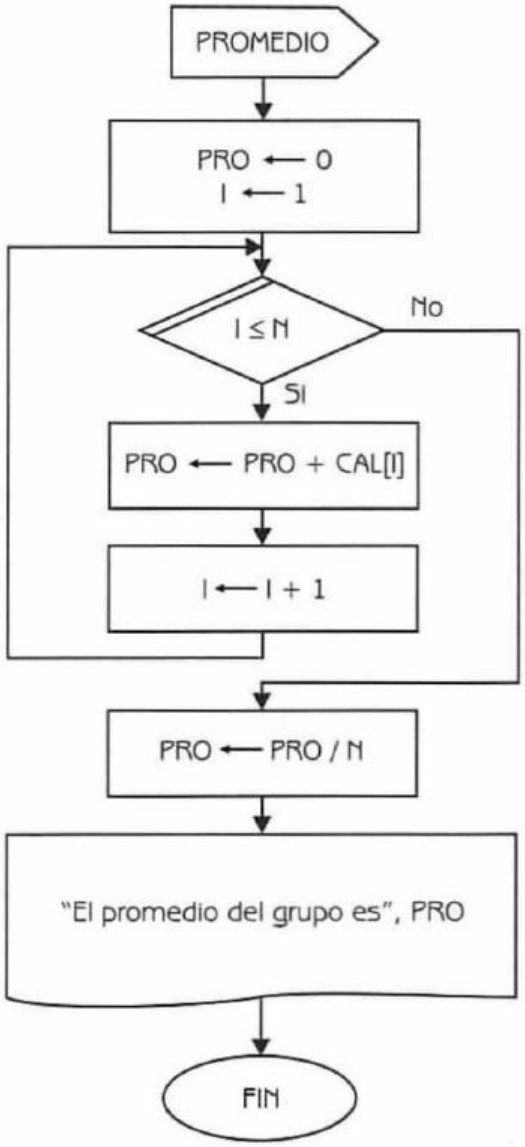


Diagrama de flujo 4.29 (*continuación*)

Explicación de variables

N: Variable de tipo entero. Indica el número de elementos que tendrán los arreglos.
I: Variable de tipo entero. Se utiliza como variable de control de los ciclos.
MAT: Arreglo unidimensional de tipo entero.
CAL: Arreglo unidimensional de tipo real.
MAC: Variable de tipo real. Se utiliza para almacenar la calificación más alta.
MAM: Variable de tipo entero. Se utiliza para almacenar la matrícula del alumno que tiene la calificación más alta.
MEC: Variable de tipo real. Se utiliza para almacenar la mayor calificación.

www.FreeLibros.com

MEM: Variable de tipo entero. Se utiliza para almacenar la matrícula del alumno con la calificación más baja.
PRO: Variable de tipo real. Almacena el promedio del grupo.

Programa 4.29

```
Arreglos –paralelos-1

{El programa trabaja sobre arreglos paralelos y permite obtener información de los
alumnos de una escuela preparatoria}.

{N, I, MAM y MEM son variables de tipo entero. MAC, MEC, y PRO son variables de
tipo real. MAT es un arreglo unidimensional de tipo entero. CAL es un arreglo unidi-
mensional de tipo real}.

1. Escribir "Ingrese el número de alumnos"
2. Leer N
3. Si (N >= 1) y (N <= 500)
   entonces
        Hacer I ←—— 1
   3.1 Repetir con I desde 1 hasta N
            Escribir "Ingrese la matrícula y la calificación del alumno", I
            Leer MAT [I], CAL [I]
            Hacer I ←—— I+1
   3.2 {Fin del ciclo del paso 3.1}
       Hacer MAC ←—— CAL [1], MAM ←—— MAT [1] e I ←—— 2
   3.3 Repetir con I desde 2 hasta N
     3.3.1 Si (MAC < CAL [I]) entonces
                Hacer MAC ←—— CAL [I] y MAM ←—— MAT [I]
     3.3.2 {Fin del condicional del paso 3.3.1}
           Hacer I ←—— I + 1
   3.4 {Fin del ciclo del paso 3.3}
       Escribir "La matrícula del alumno con mejor calificación es", MAM
       Hacer MEC ←—— CAL [1], MEM ←—— MAT [1] e I ←—— 2
   3.5 Repetir con I desde 2 hasta N
     3.5.1 Si (MEC > CAL [I]) entonces
                Hacer MEC ←—— CAL [I] y MEM ←—— MAT [I]
     3.5.2 {Fin del condicional del paso 3.5.1}
           Hacer I ←—— I + 1
   3.6 {Fin del ciclo del paso 3.5}
       Escribir "La matrícula del alumno con peor calificación es", MEM
       Hacer PRO ←—— 0 e I ←—— 1
   3.7 Repetir con I desde 1 hasta N
            Hacer PRO ←—— PRO + CAL [I] e I ←—— I + 1
```

www.FreeLibros.com

```
3.8 {Fin del ciclo del paso 3.7}
        Hacer PRO ←—— PRO / N
        Escribir "El promedio del grupo es", PRO
  sino
        Escribir "Error en el dato"
4.4 {Fin del condicional del paso 3}
```

Problema 4.22

La información sobre los productos y existencia de los mismos de la tienda "Don Marce", se almacena en dos arreglos paralelos llamados PRO y CAN. Construya un diagrama de flujo que realice lo siguiente:

a) Lea las claves y cantidades de los productos.
b) Obtenga la clave del producto que tenga la mayor existencia. Asuma que no habrá cantidades iguales. La existencia representa el número de cajas disponibles del producto.
c) Imprima la clave de los productos que se deben pedir al proveedor. Estos son aquellos cuya existencia es mayor o igual a 1 y menor a 50. Lo que se solicita es la diferencia entre 50 y la actual existencia.
d) Elimine aquellos productos cuya existencia sea cero.

Datos: PRO [1..N], CAN [1..N] $1 \leq N \leq 100$

Donde: PRO es un arreglo unidimensional de tipo entero que almacena las claves de los productos.
CAN es un arreglo unidimensional de tipo entero que almacena la existencia de los productos.

www.FreeLibros.com

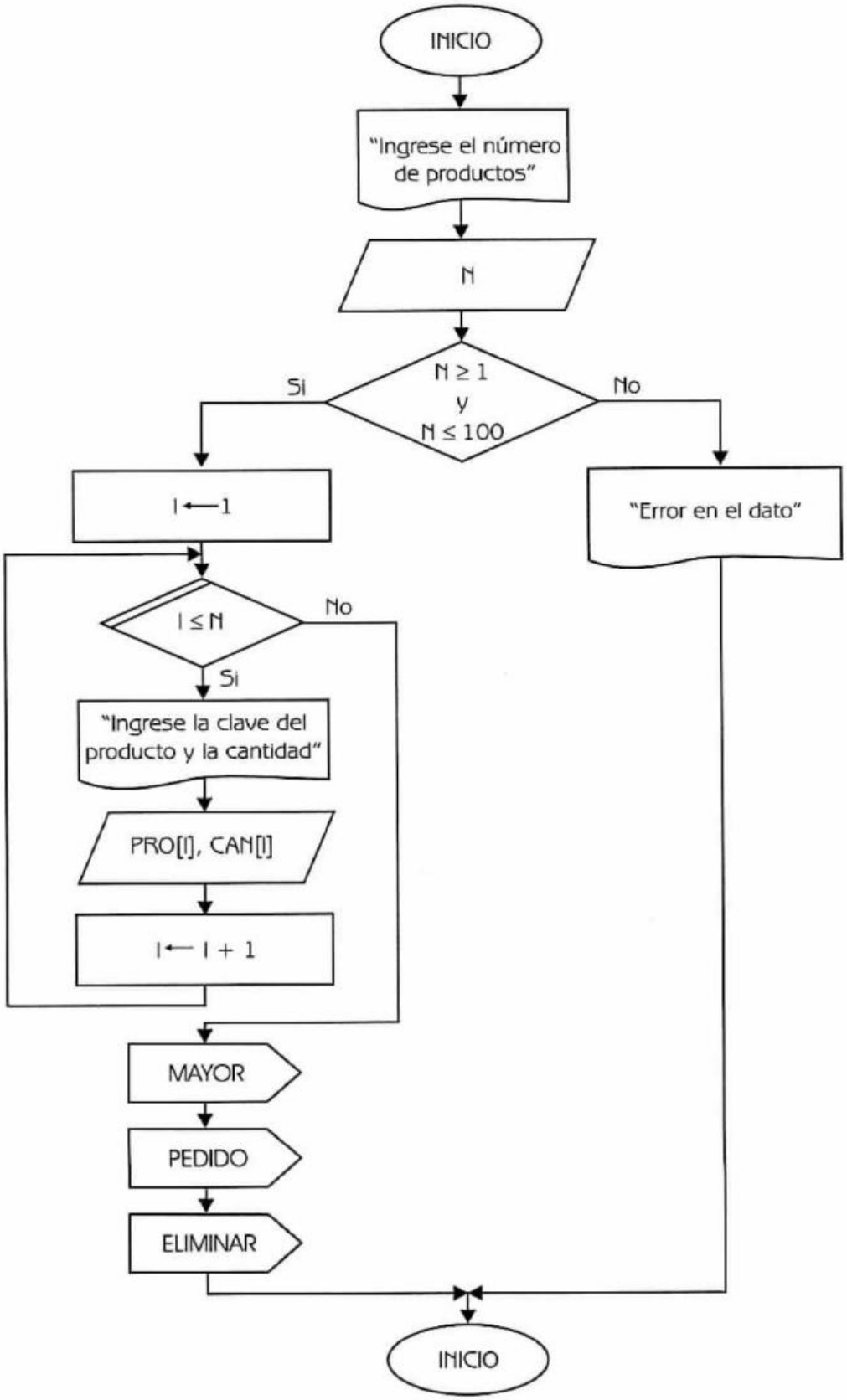


Diagrama de flujo 4.30 (*continúa*)

www.FreeLibros.com

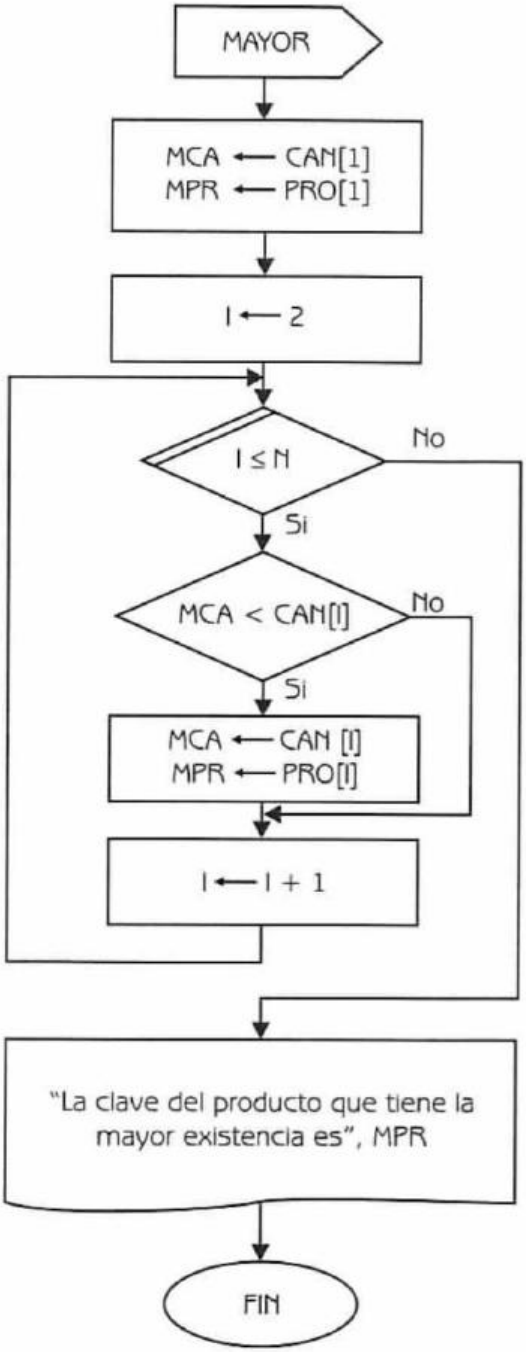


Diagrama de flujo 4.30 (*continuación...*)

www.FreeLibros.com

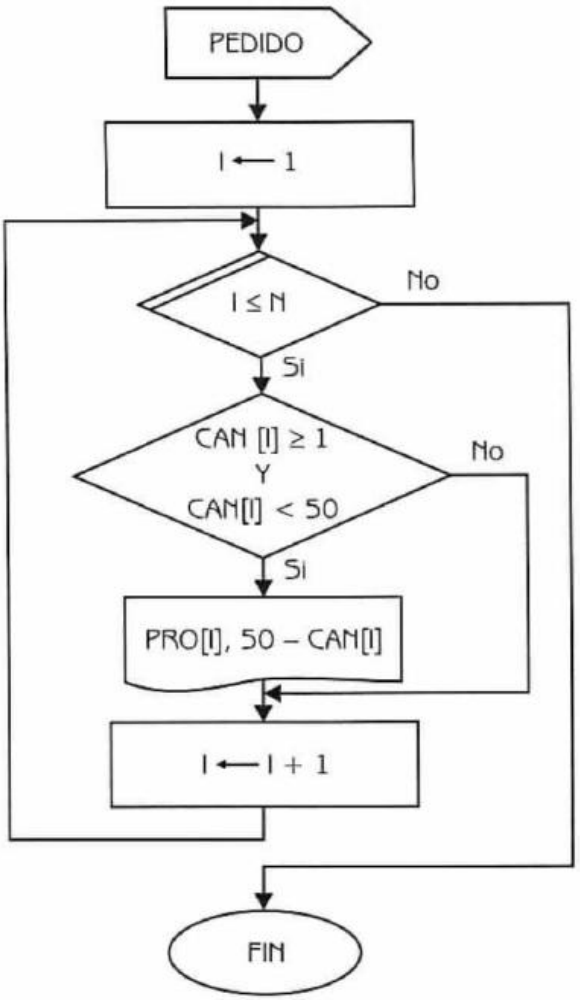


Diagrama de flujo 4.30 (*continuación...*)

www.FreeLibros.com

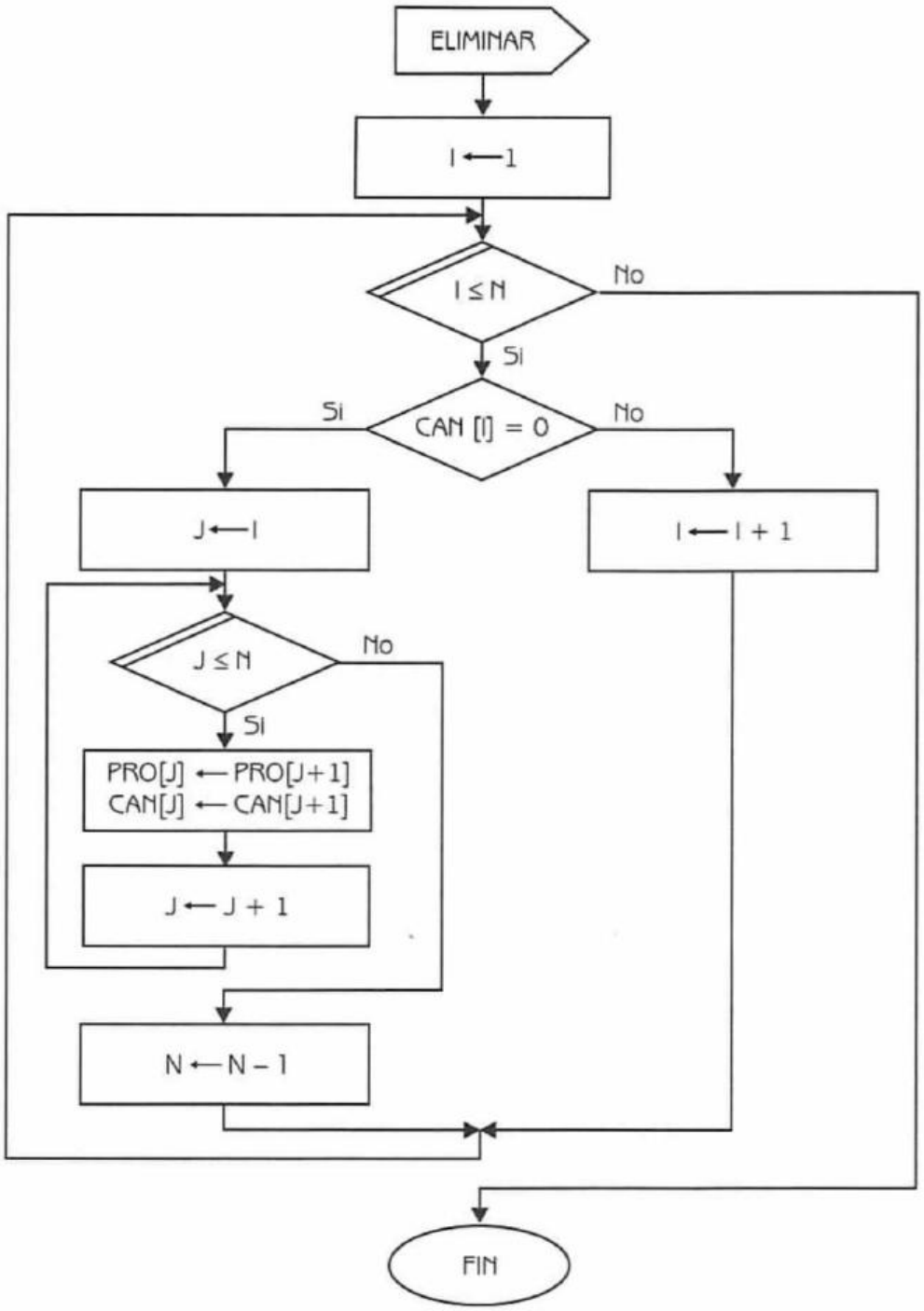


Diagrama de flujo 4.30 (*continuación*)

Explicación de variables

N: Variable de tipo entero. Indica el número de elementos que tendrán los arreglos.

I y J: Variables de tipo entero. Se utilizan como variables de control de los ciclos.

PRO y CAN: Arreglos unidimensionales de tipo entero.

MCA: Variable de tipo entero. Se utiliza para almacenar la existencia más alta.

www.FreeLibros.com

MPR: Variable de tipo entero. Se utiliza para almacenar la clave del producto que tiene la existencia más alta.

Programa 4.30

```
Arreglos-paralelos-2

{El programa trabaja sobre arreglos paralelos y permite obtener información de los
productos que vende la tienda "Don Marce"}

{N, I, J, MCA y MPR son variables de tipo entero. PRO y CAN son arreglos unidimen-
sionales de tipo entero}.

1. Escribir "Ingrese el número de productos"
2. Leer N
3. Si (N >= 1) y (N <= 100)
   entonces
         Hacer I ←—— 1
   3.1 Repetir con I desde 1 hasta N
            Escribir "Ingrese la clave del producto y la cantidad"
            Leer PRO [I] y CAN [I]
            Hacer I ←—— I +1
   3.2 {Fin del ciclo del paso 3.1}
            Hacer MCA ←—— CAN [1], MPR ←—— PRO [1] e I ←—— 2
   3.3 Repetir con I desde 2 hasta N
     3.3.1 Si MCA < CAN [I] entonces
                  Hacer MCA ←—— CAN [I] y MPR ←—— PRO [I]
     3.3.2 {Fin del condicional del paso 3.3.1}
                  Hacer I ←—— I + 1
   3.4 {Fin del ciclo del paso 3.3}
            Escribir "La clave del producto que tiene la mayor existencia es", MPR
            Hacer I ←—— 1
   3.5 Repetir con I desde 1 hasta N
     3.5.1 Si (CAN [I] >= 1) y (CAN [I] < 50) entonces
                  Escribir PRO [I] y 50 – CAN [I]
     3.5.2 {Fin del condicional del paso 3.5.1}
                  Hacer I ←—— I + 1
   3.6 {Fin del ciclo del paso 3.5}
            Hacer I ←—— 1
   3.7 Mientras (I <= N) Repetir
     3.7.1 Si CAN [I] = 0
                  entonces
                        Hacer J ←—— I
```

www.FreeLibros.com

```
            3.7.1.1 Repetir con J desde I hasta N
                        Hacer PRO [J] ⟵ PRO [J + 1], CAN [J] ⟵ CAN [J + 1]
                                y J ⟵ J + 1
            3.7.1.2 {Fin del ciclo del paso 3.7.1.1}
                        Hacer N ⟵ N – 1
                    sino
                        Hacer I ⟵ I + 1
        3.7.2 {Fin del condicional del paso 3.7.1}
    3.8 {Fin del ciclo del paso 3.7}
sino
                Escribir "Error en el dato"
4. {Fin del condicional del paso 3}
```

www.FreeLibros.com

5

Estructuras de datos: registros

5.1 Registros

Cuando hablamos de arreglos mencionamos que se trataba de una colección de datos y que con ellos se podían solucionar un gran número de problemas. Sin embargo, en la práctica a veces se necesitan estructuras que permitan almacenar diferentes tipos de datos bajo un mismo nombre. Para ilustrar este problema se incluye el siguiente ejemplo:

Ejemplo 5.1

Consideremos que una escuela tiene por cada alumno los siguientes datos:

- Nombre (cadena de caracteres).
- Dirección (cadena de caracteres).
- Matrícula (entero).
- Carrera (cadena de caracteres).

Si se quisiera almacenar estos datos no sería posible utilizar un arreglo, ya que sus componentes deben ser todos del mismo tipo. La estructura que permite almacenar esta información es la que se conoce con el nombre de *registro*.

Un *registro* es un tipo de dato estructurado, donde cada uno de sus componentes se denomina campo. Los campos de un registro pueden ser todos de diferentes tipos. Por lo tanto, también podrán ser registros o arreglos. Cada campo se identifica por un nombre único (el indentificador de campo).

www.FreeLibros.com

5.1.1 Definición de registros

Como no es nuestra intención seguir la sintaxis de algún lenguaje de programación, definimos un registro de la siguiente manera:

```
ident_registro = REGISTRO
    id_campo1: tipo1
    id_campo2: tipo2
    . . .

    id_campon: tipon
{Fin de la definición del registro}
```

Donde:

ident_registro	representa el nombre del dato tipo registro.
id_campo_i	es el nombre del campo i.
$tipo_i$	es el tipo de datos del campo i.

Veamos a continuación algunos ejemplos de definición de registros y su correspondiente representación gráfica.

Ejemplo 5.2

Sea TIEMPO un registro formado por tres campos numéricos. Su representación queda como se muestra en la figura 5.1.

```
TIEMPO = REGISTRO
    hora: 1..24
    minutos: 0..59
    segundos: 0..59
{Fin de la definición del registro TIEMPO}
```

Figura 5.1

www.FreeLibros.com

Ejemplo 5.3

Sea DOMICILIO un registro formado por cuatro campos, uno de los cuales es numérico y los tres restantes son del tipo cadena de caracteres. Su representación queda como se muestra en la figura 5.2.

```
DOMICILIO = REGISTRO
    calle: cadena de caracteres
    número: entero
    ciudad: cadena de caracteres
    país: cadena de caracteres
{Fin de la definición del registro DOMICILIO}
```

Figura 5.2

Ejemplo 5.4

Sea EMPLEADO un registro formado por cinco campos.

- Dos del tipo cadena de caracteres.
- Dos del tipo entero.
- Uno del tipo real.

Su representación queda como se muestra en la figura 5.3.

```
EMPLEADO = REGISTRO
    número: entero
    nombre: cadena de caracteres
    departamento: cadena de caracteres
    nivel: entero
    sueldo: real
{Fin de la definición del registro EMPLEADO}
```

Figura 5.3

www.FreeLibros.com

5.1.2 Acceso a los campos de un registro

Como un registro es un dato estructurado no puede accesarse directamente como un todo. Se debe especificar qué elemento (campo) del registro interesa. Para ello se sigue esta sintaxis:

variable_registro.id_campo

Donde:

variable_registro es una variable de tipo registro.

id_campo es el identificador del campo deseado.

Es decir, se usarán dos nombres para hacer referencia a un elemento: el nombre de la variable tipo registro y el nombre del componente, separados entre sí por un punto. Los siguientes casos ilustran el acceso a campos de registros.

a) Para leer los tres campos de una variable T de tipo TIEMPO hacemos:

Leer (T.hora, T.minutos, T.segundos)

b) Para escribir los cuatro campos de una variable D de tipo DOMICILIO hacemos:

Escribir (D.calle, D.número, D.ciudad, D.país)

c) Para asignar valores a algunos de los campos de una variable E de tipo EMPLEADO hacemos:

E.número ← 120
E.nombre ← "Ricardo Gómez"
E.departamento ← "Contabilidad"
E.nivel ← 3
E.sueldo ← 1280.50

En general el orden en el cual se manejan los campos no es importante. Es decir, se podrían haber leído los campos de la variable T de la siguiente manera:

Leer(T.segundos, T.minutos, T.hora)

www.FreeLibros.com

5.1.3 Diferencias con arreglos

Las tres diferencias sustanciales existentes con la estructura de datos tipo arreglo son las siguientes:

1. Un arreglo puede almacenar N elementos del mismo tipo, mientras que un registro puede almacenar N elementos de diferentes tipos.
2. Los componentes de un arreglo se accesan por medio de índices (que indican la posición del componente en el arreglo), mientras que los componentes de un registro (los campos) se accesan por medio de su nombre, el cual es único.
3. En un arreglo es posible establecer un orden entre sus elementos, mientras que en un registro no existe orden entre los campos.

5.1.4 Combinaciones entre arreglos y registros

Anteriormente mencionamos que un campo de un registro puede ser de cualquier tipo, por lo tanto puede ser un registro o un arreglo. A su vez cada componente de un arreglo puede ser un registro. La siguiente figura ilustra las combinaciones que pueden presentarse entre estas estructuras.

Figura 5.4 Combinaciones entre arreglos y registros.

Arreglos de registros

En este caso cada elemento del arreglo es un registro (todos los componentes del arreglo serán del mismo tipo de registro). A continuación presentamos un ejemplo.

Ejemplo 5.5

Una empresa registra para cada uno de sus empleados los siguientes datos:

- NUM (entero).
- NOM (cadena de caracteres).
- DEP (cadena de caracteres).
- NIV (entero).
- SUE (real).

www.FreeLibros.com

Donde:

NUM	representa el número del empleado.
NOM	representa el nombre del empleado.
DEP	indica el departamento en que trabaja el empleado.
NIV	indica el nivel del trabajador.
SUE	representa el sueldo del empleado.

Suponiendo que la empresa tiene N empleados necesitaremos entonces un arreglo de N elementos, en el cual cada componente del mismo es un registro del tipo descrito en el ejemplo 5.4.

EMP = ARREGLO [1..50] DE EMPLEADO

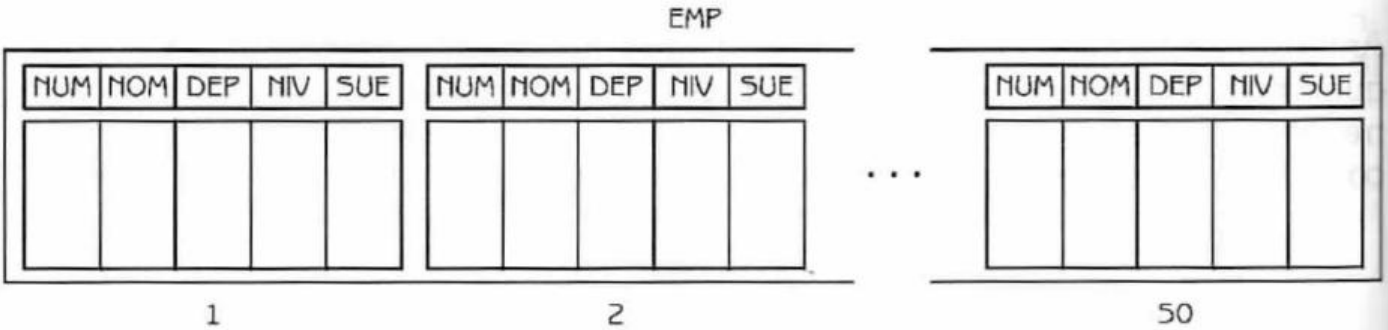
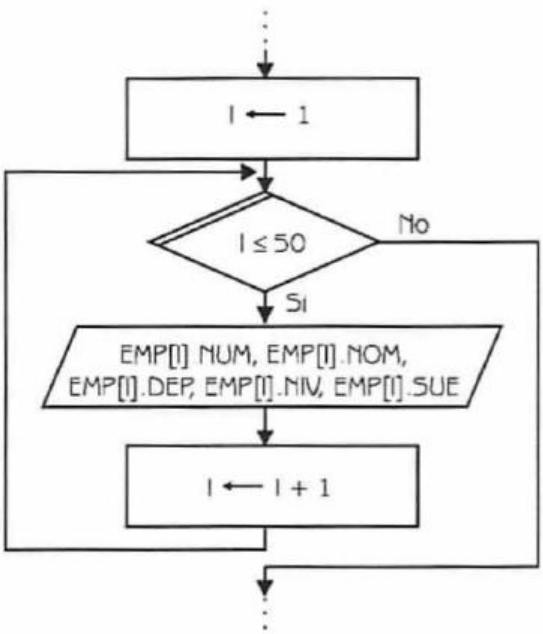


Figura 5.5 Arreglos de registros.

Cada elemento de EMP será un dato tipo EMPLEADO. Por lo tanto si se quiere por ejemplo leer el arreglo EMP, debe leerse por cada componente cada uno de los campos que forman el registro.



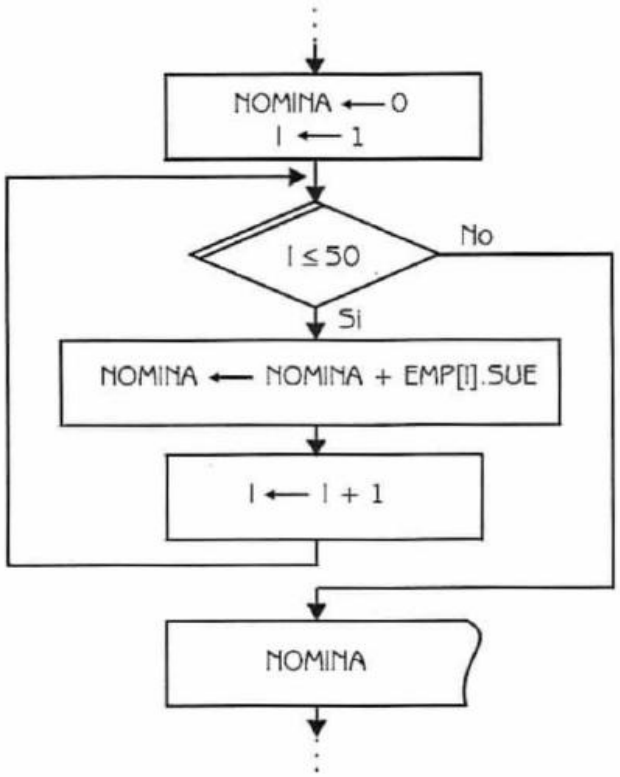
www.FreeLibros.com

```
.
.
Hacer I ←—— 1
Repetir con I desde 1 hasta 50
   Leer EMP[I]. NUM, EMP[I]. NOM, EMP[I]. DEP, EMP[I]. NIV, EMP[I]. SUE
   Hacer I ←—— I + 1
{Fin del ciclo}
.
.
```

Con EMP[I] se hace referencia al elemento I del arreglo EMP, que es un registro; con id_campo se especifica cuál de los campos del registro es el que se leerá. De manera similar procedemos para escritura, asignación o comparación de elementos.

Supongamos ahora que la empresa necesita obtener la siguiente información:

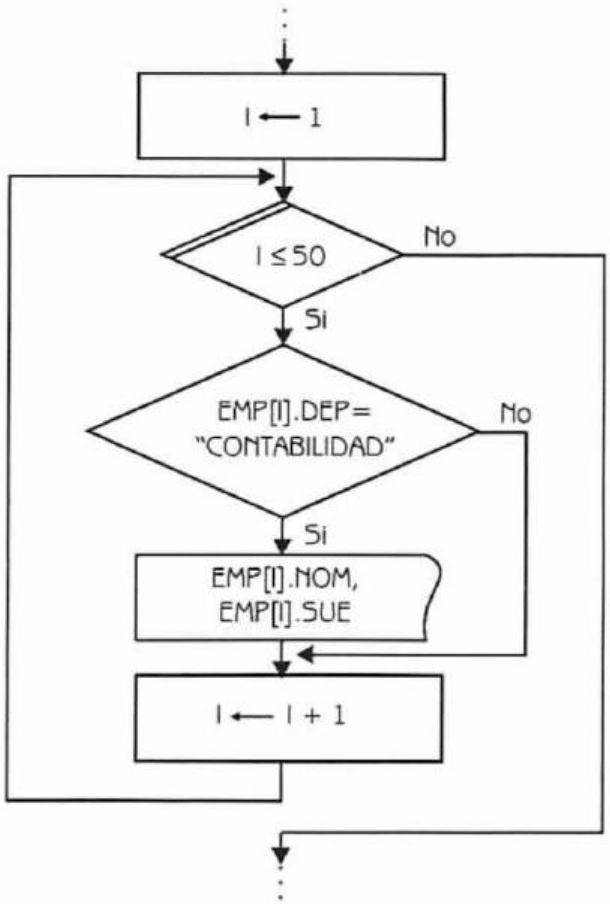
a) El total de nómina, es decir el total de lo que ganan sus empleados.




```
.
.
Hacer NOMINA ←—— 0 e I ←—— 1
Repetir con I desde 1 hasta 50
      Hacer NOMINA ←—— NOMINA + EMP[I]. SUE e I ←—— I + 1
{Fin del ciclo}
Escribir NOMINA
.
.
```

www.FreeLibros.com

b) El nombre y el sueldo de los empleados del departamento de contabilidad.




```
.
.
Hacer I ←— 1
Repetir con I desde 1 hasta 50
      Si EMP[I]. DEP = "Contabilidad" entonces
            Escribir EMP[I].NOM y EMP[I]. SUE
      {Fin del condicional}
      Hacer I ←— I + 1
{Fin del ciclo}
.
.
.
```

Registros anidados

En los registros anidados, al menos un campo del registro es del tipo registro. Veamos a continuación un ejemplo.

Ejemplo 5.6

Una empresa registra para cada uno de sus empleados los siguientes datos:

- Número (entero).
- Nombre (cadena de caracteres).

www.FreeLibros.com

- Dirección : Calle (cadena de caracteres).
  Número (entero).
  Ciudad (cadena de caracteres).
  País (cadena de caracteres).
- Departamento(cadena de caracteres).
- Nivel (entero).
- Sueldo (real).

Para precisar el tipo del campo Dirección es necesario definir previamente un registro formado por los cuatro componentes (calle, número, ciudad y país) que se especifican.

```
EMPLEADO = REGISTRO
      número: entero
      nombre: cadena de caracteres
      dirección: DOMICILIO
      departamento: cadena de caracteres
      nivel: entero
      sueldo: real
{Fin de la definición del registro EMPLEADO}
```

La figura 5.6 muestra la estructura de datos requerida.

Figura 5.6 Registros anidados.

En este caso el registro tiene un campo (dirección) que es del tipo DOMICILIO, el cual es un registro de cuatro campos.

Para tener acceso a los campos que a su vez son registros, en la mayoría de los lenguajes se sigue la siguiente sintaxis:

Variable_registro.id_campo$_i$.id_campo$_j$

Donde:

variable_registro es una variable de tipo registro.
id_campo$_i$ es un identificador de un campo del registro (el campo es de tipo registro).

www.FreeLibros.com

id_campoj es el identificador de un campo del segundo registro.

Para accesar los campos del registro del ejemplo anterior, la secuencia a seguir es la siguiente:

EMPLEADO.número
EMPLEADO.nombre
EMPLEADO.dirección.calle
EMPLEADO.dirección.número
EMPLEADO.dirección.ciudad
EMPLEADO.dirección.país
EMPLEADO.departamento
EMPLEADO.nivel
EMPLEADO.sueldo

Registros con arreglos

Los registros con arreglos tienen por lo menos un campo que es un arreglo. Observemos el siguiente ejemplo.

Ejemplo 5.7

El Centro Meteorológico de México registra para cada uno de los estados los siguientes datos:

- Nombre (cadena de caracteres).
- LLuvias mensuales del último año (arreglo unidimensional).

La definición del registro correspondiente es la siguiente:

```
LLUVIAS = REGISTRO
          nombre: cadena de caracteres
          llumen: ARREGLO [1..12] DE reales
{Fin de la definición del registro LLUVIAS}
```

La figura 5.7 muestra la estructura requerida:

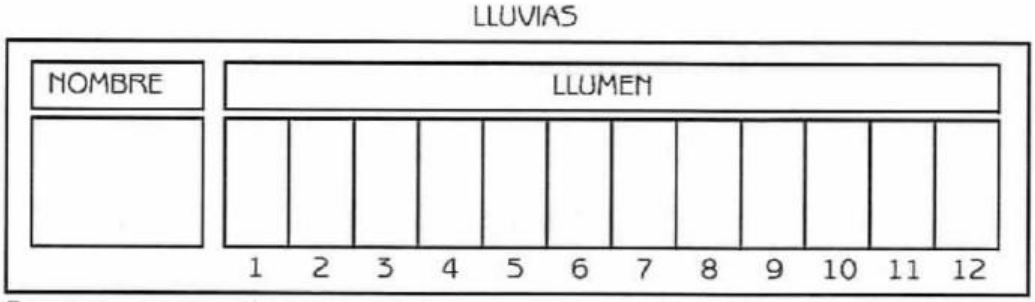


Figura 5.7 Registros con arreglos.

www.FreeLibros.com

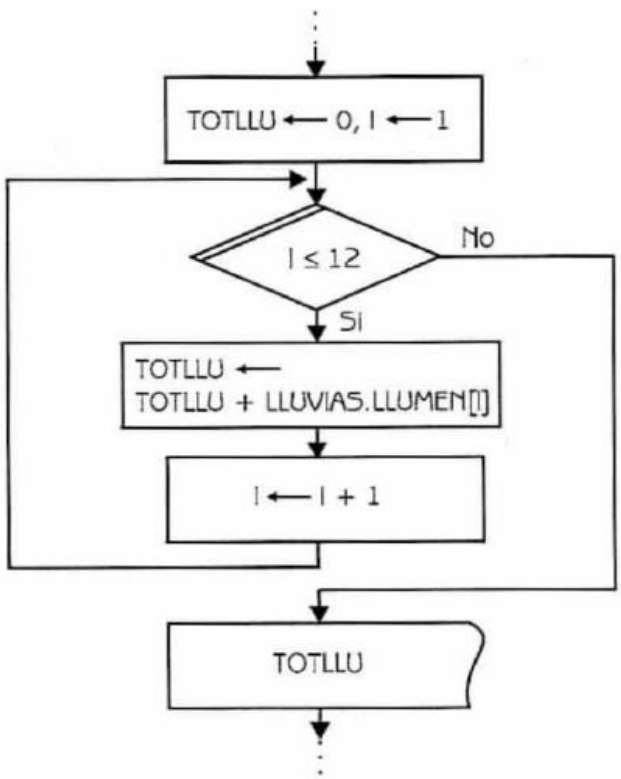
En este caso el registro tiene un campo (llumen) que es un arreglo de doce elementos reales. Para hacer referencia a ese campo, se procederá de la siguiente manera:

variable_registro.id_campo[índice]

Para accesar los campos del registro seguimos la siguiente secuencia:

```
LLUVIAS.NOMBRE
Repetir con I desde 1 hasta 12
      LLUVIAS.LLUMEN[I]
{Fin del ciclo}
```

Consideremos ahora que el Centro Meteorológico necesita obtener el total anual de lluvias de un estado. Procederemos de la siguiente manera:




```
.
.
.
Hacer TOTLLU ←— 0 e I ←— 1
Repetir con I desde 1 hasta 12
  Hacer TOTLLU ←—TOTLLU + LLUVIAS.LLUMEN[I] e I←— I + 1
{Fin del ciclo}
Escribir TOTLLU
.
.
.
```

A continuación presentamos una serie de problemas resueltos, diseñados expresamente como elementos de ayuda para el análisis y retención de los conceptos.

www.FreeLibros.com

Problemas resueltos

Problema 5.1

Una matriz esparcida se caracteriza porque la mayoría de sus elementos son ceros. No resulta conveniente por lo tanto almacenar todos los elementos, sino solamente aquellos que son distintos de cero. Para ello se procede de la siguiente manera: "Cada valor distinto de cero se almacena junto con el renglón y la columna en el cual se localiza".

Escriba un diagrama de flujo que permita llevar a cabo las siguientes operaciones:

a) Leer los elementos correspondientes a dos matrices esparcidas A (M x N) y B (M x N) y almacenar los valores distintos de 0 en los arreglos unidimensionales A1 y B1.

b) Obtener una matriz C (M x N) como la suma de las matrices A y B, pero utilizando los arreglos A1 y B1.

c) Imprimir la matriz C.

Datos: $A_{1,1}, A_{1,2}, \ldots, A_{1,N}$ $\quad$ $B_{1,1}, B_{1,2} \ldots, B_{1,N}$
$A_{2,1}, A_{2,2}, \ldots, A_{2,N}$ $\quad$ $B_{2,1}, B_{2,2} \ldots, B_{2,N}$
$\ldots$ $\quad$ $\ldots$
$A_{M,1}, A_{M,2}, \ldots, A_{M,N}$ $\quad$ $B_{M,1}, B_{M,2} \ldots, B_{M,N}$

$1 \leq M \leq 50$ $\quad$ $1 \leq N \leq 30$

Donde: $A_{i,j}$ y $B_{i,j}$ son variables de tipo entero que representan elementos de las matrices A y B, respectivamente ($1 \leq i \leq M$, $1 \leq j \leq N$).

www.FreeLibros.com

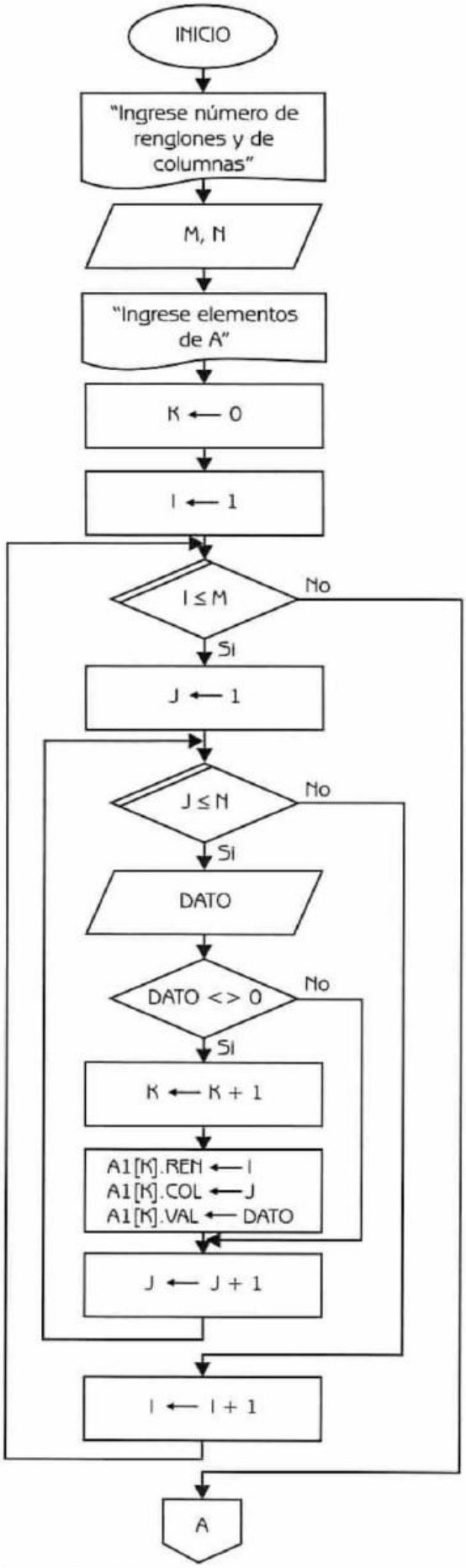
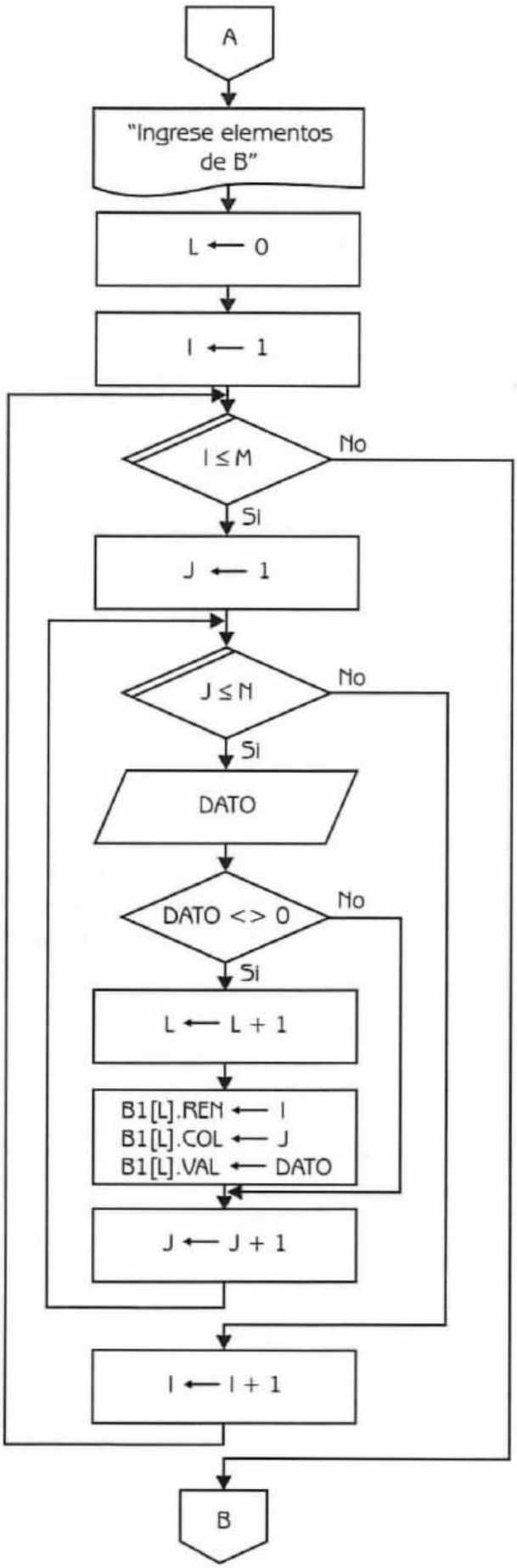


Diagrama de flujo 5.1 (*continúa*)

Nota:
Se leen los elementos de la matriz A y se almacenan los elementos distintos de cero en el arreglo unidimensional A1.

www.FreeLibros.com



Nota:
Se leen los elementos de la matriz B y se almacenan los elementos distintos de cero en el arreglo unidimensional B1.

Diagrama de flujo 5.1 *(continuación...)*

www.FreeLibros.com

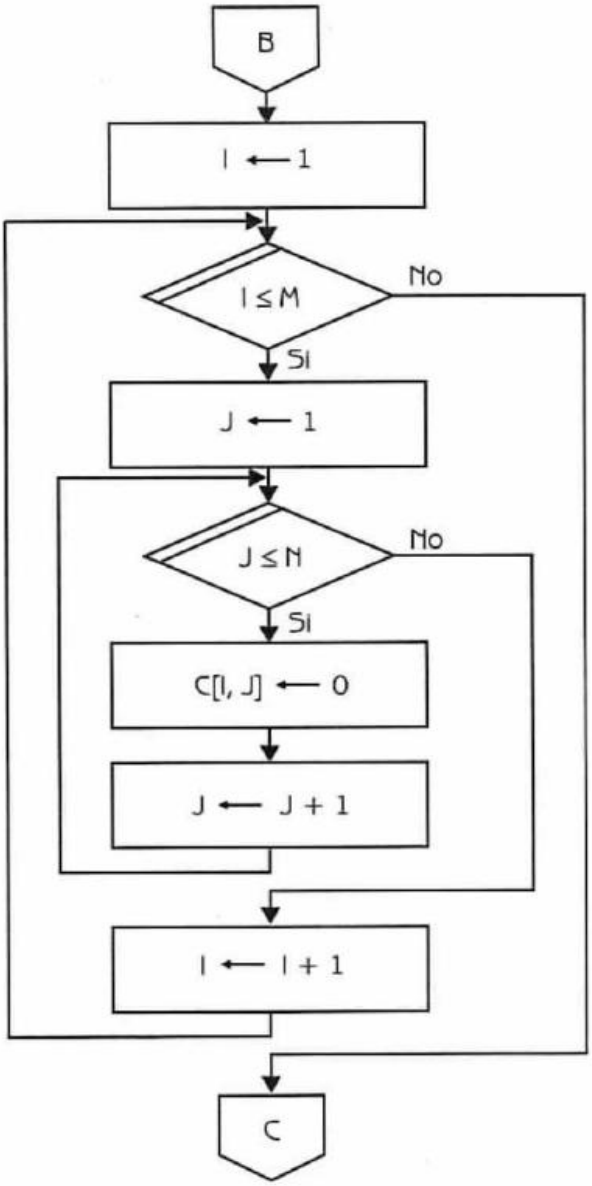
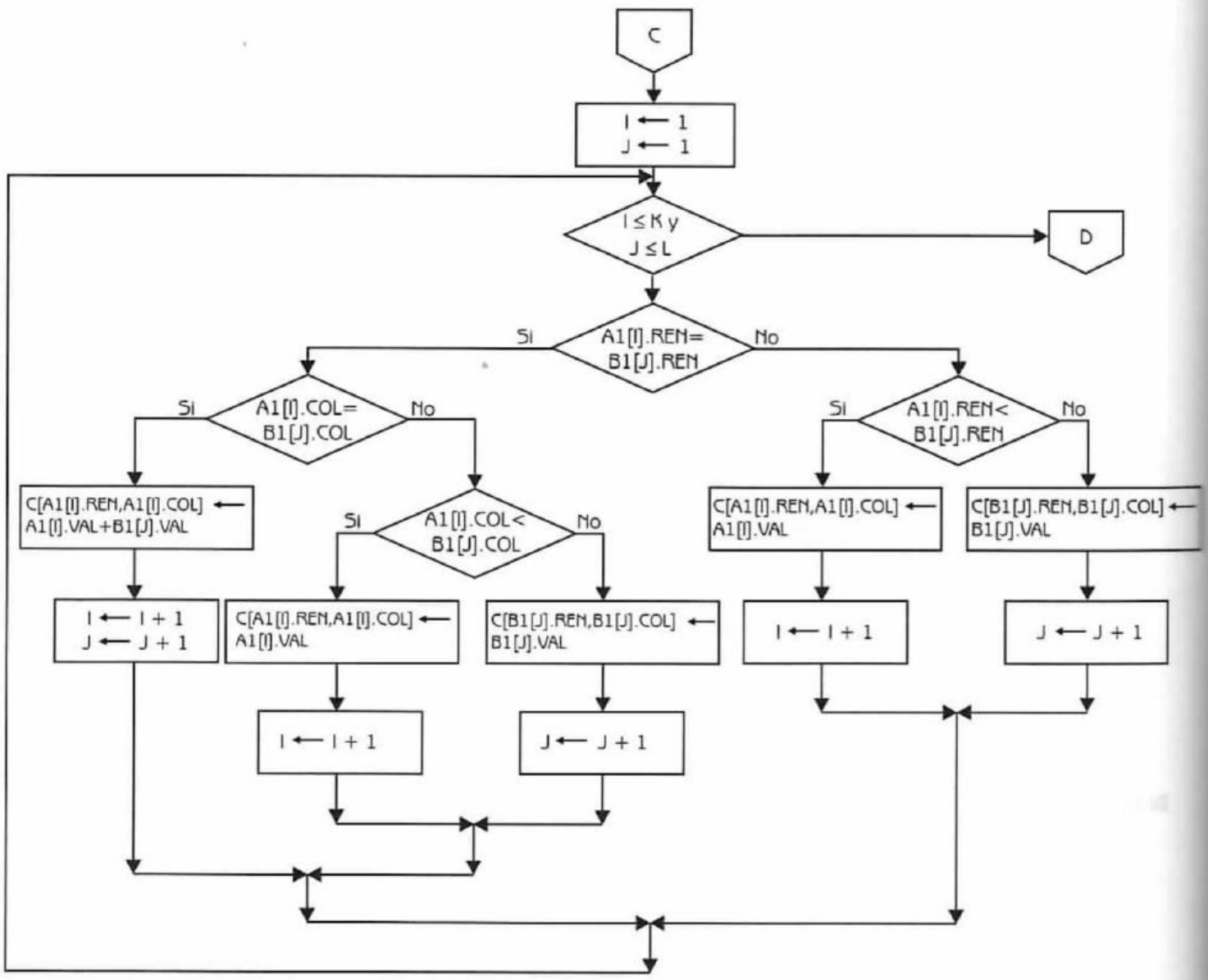


Diagrama de flujo 5.1 (*continuación...*)

Nota:
Se inicializa la matriz C.

www.FreeLibros.com



Nota:
Se almacena la suma de las matrices A y B, que se encuentran almacenadas en los arreglos unidimensionales A1 y B1, en la matriz C.

Diagrama de flujo 5.1 (*continuación...*)

www.FreeLibros.com

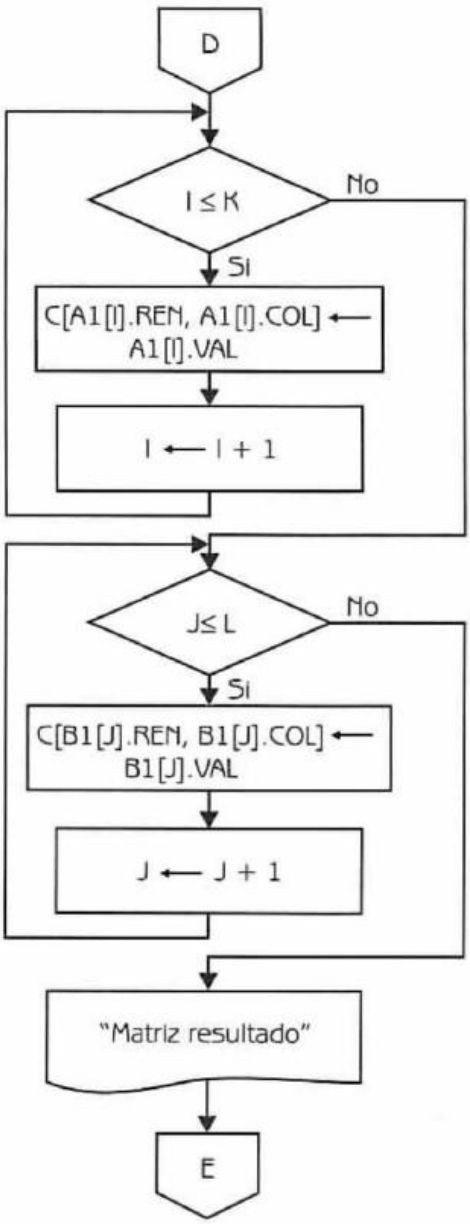
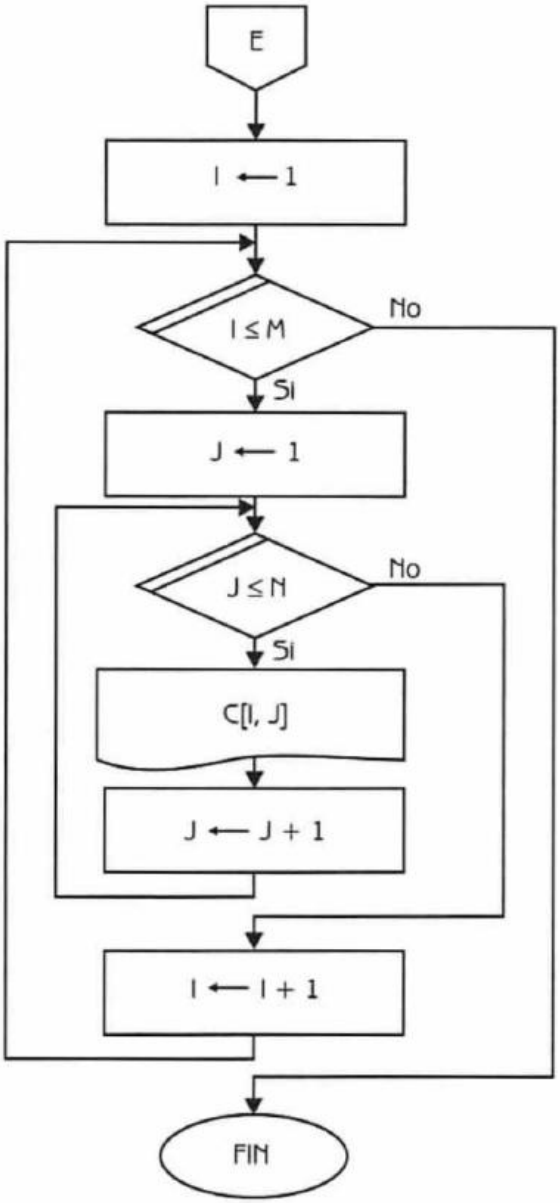


Diagrama de flujo 5.1 (*continuación...*)

www.FreeLibros.com



Nota:
Escritura de la matriz C, que contiene la suma de A y B.

Diagrama de flujo 5.1 *(continuación)*

Explicación de las variables

M, N: Variables de tipo entero. Almacenan las dimensiones de los arreglos.

I, J: Variables de tipo entero. Se usan como índices de los arreglos y como variables de control de los ciclos.

K y *L*: Variables de tipo entero. Almacenan el total de valores diferentes de cero correspondientes a los arreglos A y B respectivamente.

A1 y *B1*: Arreglos unidimensionales de **tipo registro**. Cada elemento de los arreglos almacena un valor diferente de cero, junto con el número de renglón y el número de columna, correspondiente a su posición original en los arreglos A y B respectivamente.

www.FreeLibros.com

DATO: Variable de tipo entero. Se usa para leer los elementos de las matrices.

C: Arreglo bidimensional de tipo entero. Almacena la suma de los arreglos A y B (almacenados a su vez en A1 y B1 respectivamente)

A continuación presentamos el programa correspondiente.

Programa 5.1

```
MATRICES_ESPARCIDAS

{El programa almacena los elementos de dos matrices esparcidas en dos arreglos
unidimensionales de registros. Posteriormente se obtiene una matriz que repre-
senta la suma de las matrices esparcidas}

{M, N, I, J, K, L y DATO son variables de tipo entero. A1 y B1 son arreglos unidimen-
sionales de tipo registro. A, B y C son arreglos bidimensionales de tipo entero}

1. Escribir " Ingrese número de renglones y de columnas"
2. Leer M y N
3. Escribir " Ingrese elementos de A"
4. Hacer K ←—— 0 e I ←—— 1
5. Repetir con I desde 1 hasta M
        Hacer J ←—— 1
   5.1 Repetir con J desde 1 hasta N
            Leer DATO
       5.1.1 Si DATO <> 0 entonces
                Hacer K ←—— K+1, A1[K].REN ←—— I,
                    A1[K].COL ←—— J y A1[K].VAL ←—— DATO
       5.1.2 {Fin del condicional del paso 5.1.1}
            Hacer J ←—— J+1
   5.2 {Fin del ciclo del paso 5.1}
       Hacer I ←—— I+1
6. {Fin del ciclo del paso 5}
7. Escribir " Ingrese elementos de B"
8. Hacer L ←—— 0 e I ←—— 1
9. Repetir con I desde 1 hasta M
        Hacer J ←—— 1
   9.1 Repetir con J desde 1 hasta N
            Leer DATO
       9.1.1 Si DATO <> 0 entonces
```

www.FreeLibros.com

```
                    Hacer L ⟵ L+1, B1[L].REN ⟵ I,
                            B1[L].COL ⟵ J y B1[L].VAL ⟵ DATO
        9.1.2 {Fin del condicional del paso 9.1.1}
              Hacer J ⟵ J+1
    9.2 {Fin del ciclo del paso 9.1}
        Hacer I ⟵ I+1
10. {Fin del ciclo del paso 9}
11. Hacer I ⟵ 1
12. Repetir con I desde 1 hasta M
        Hacer J ⟵ 1
    12.1 Repetir con J desde 1 hasta N
              Hacer C[I, J] ⟵ 0 y J ⟵ J+1
    12.2 {Fin del ciclo del paso 12.1}
        Hacer I ⟵ I+1
13. {Fin del ciclo del paso 12
14. Hacer I ⟵ 1 y J ⟵ 1
15. Mientras (I <= K) y (J <= L) repetir
    15.1 Si A1[I].REN = B1[J].REN
              entonces
         15.1.1 Si A1[I].COL = B1[J].COL
                  entonces
                        Hacer C[ A1[I].REN, A1[I].COL] ⟵
                              A1[I].VAL + B1[J].VAL, I ⟵ I+1
                              y J ⟵ J+1
                  sino
                  15.1.1.1 Si A1[I].COL < B1[J].COL
                              entonces
                                    Hacer C[ A1[I].REN, A1[I].COL] ⟵
                                          A1[I].VAL e I ⟵ I+1
                              sino
                                    Hacer C[ B1[J].REN, B1[J].COL] ⟵
                                          B1[J].VAL y J ⟵ J+1
                  15.1.1.2 {Fin del condicional del paso 15.1.1.1}
         15.1.2 {Fin del condicional del paso 15.1.1}
                 sino
         15.1.3 Si A1[I].REN < B1[J].REN
                        entonces
                              Hacer C[ A1[I].REN, A1[I].COL] ⟵
                                    A1[I].VAL e I ⟵ I+1
                        sino
                              Hacer C[ B1[J].REN, B1[J].COL] ⟵
                                    B1[J].VAL y J ⟵ J+1
```

www.FreeLibros.com

```
                15.1.4 {Fin del condicional del paso 15.1.3}
    15.2 {Fin del condicional del paso 15.1}
16. {Fin del ciclo del paso 15}
17. Mientras (I <= K) repetir
        Hacer C[ A1[I].REN, A1[I].COL] ⟵ A1[I].VAL e I ⟵ I+1
18. {Fin del ciclo del paso 17}
19. Mientras (J <= L) repetir
        Hacer C[ B1[J].REN, B1[J].COL] ⟵ B1[J].VAL y J ⟵ J+1
20. {Fin del ciclo del paso 19}
21. Escribir " Matriz resultado"
22. Hacer I ⟵ 1
23. Repetir con I desde 1 hasta M
        Hacer J ⟵ 1
    23.1 Repetir con J desde 1 hasta N
            Escribir C[I, J] y J ⟵ J+1
    23.2 {Fin del ciclo del paso 23.1}
        Hacer I ⟵ I+1
24. {Fin del ciclo del paso 23}
```

Problema 5.2

En una tienda importante de la Ciudad de México se almacena información, relativa a las ventas efectuadas por sus empleados en los 12 meses del año anterior. Esta información se almacena en un arreglo de registros con la siguiente estructura.

Observe el lector que cada elemento del arreglo "empleados" es un registro, que contiene la siguiente información:

NUMERO: Entero.

NOMBRE: Cadena de caracteres.

www.FreeLibros.com

VENTAS: Arreglo unidimensional de tipo real. Su capacidad máxima es para 12 elementos.

SALARIO: Real.

Construya un diagrama de flujo que realice las siguientes operaciones:

a) Listar el número y nombre del empleado que tuvo las mayores ventas durante el año.

b) Incremente un 10% el salario de todos los empleados que durante el año obtuvieron ventas superiores a $1'000,000.

c) Liste el número y nombre de los empleados que en el mes de diciembre tuvieron ventas inferiores a $30,000.

Dato: EMPLEADOS [1..N] $1 \leq N \leq 100$

Donde:

EMPLEADOS	es un arreglo unidimensional de registros.
NUMERO, NOMBRE, VENTAS y SALARIO	son campos del registro de tipo entero, cadena de caracteres, arreglo unidimensional de tipo real y real, respectivamente.

www.FreeLibros.com

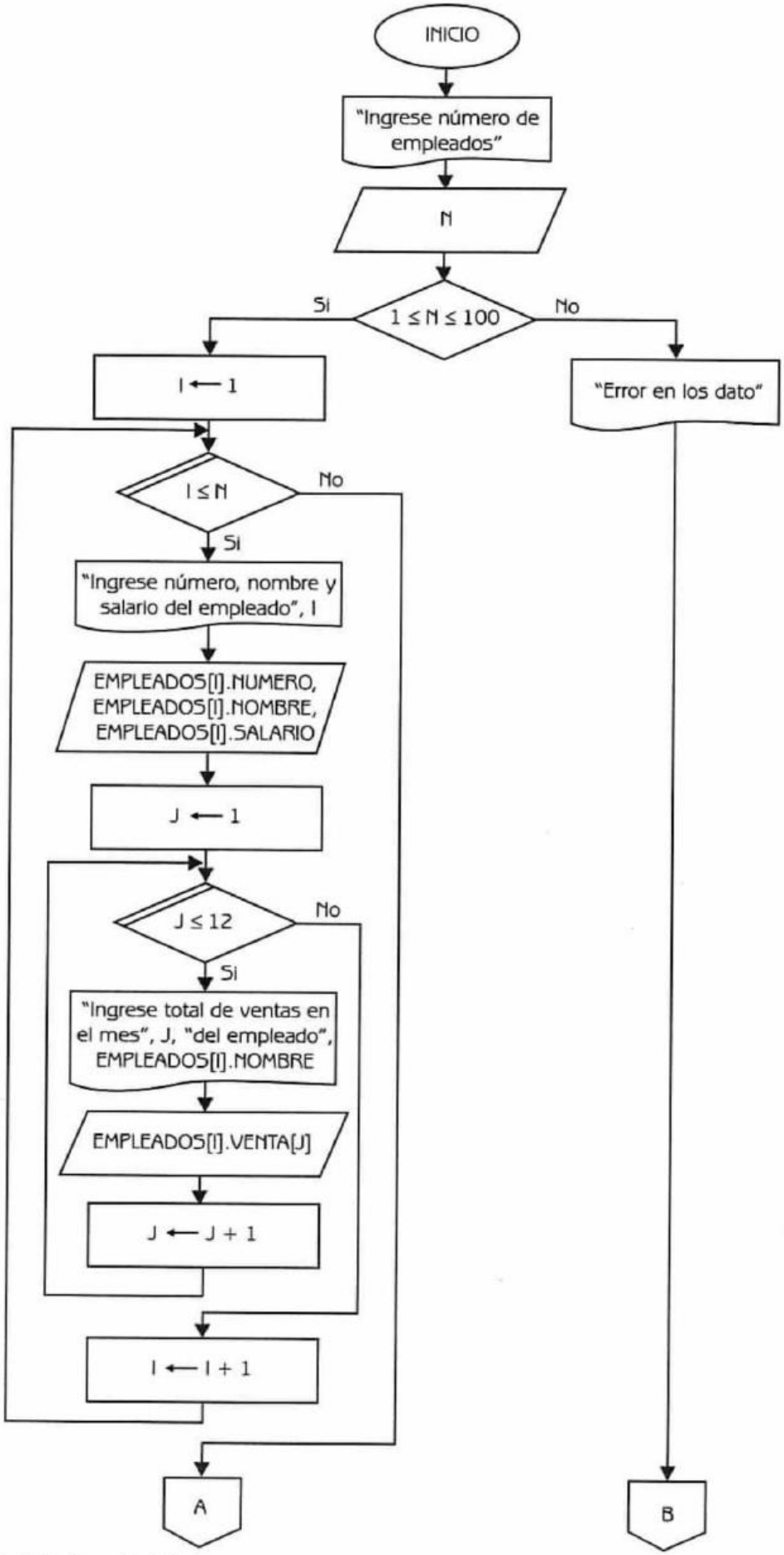


Diagrama de Flujo 5.2 (*continúa*)

www.FreeLibros.com

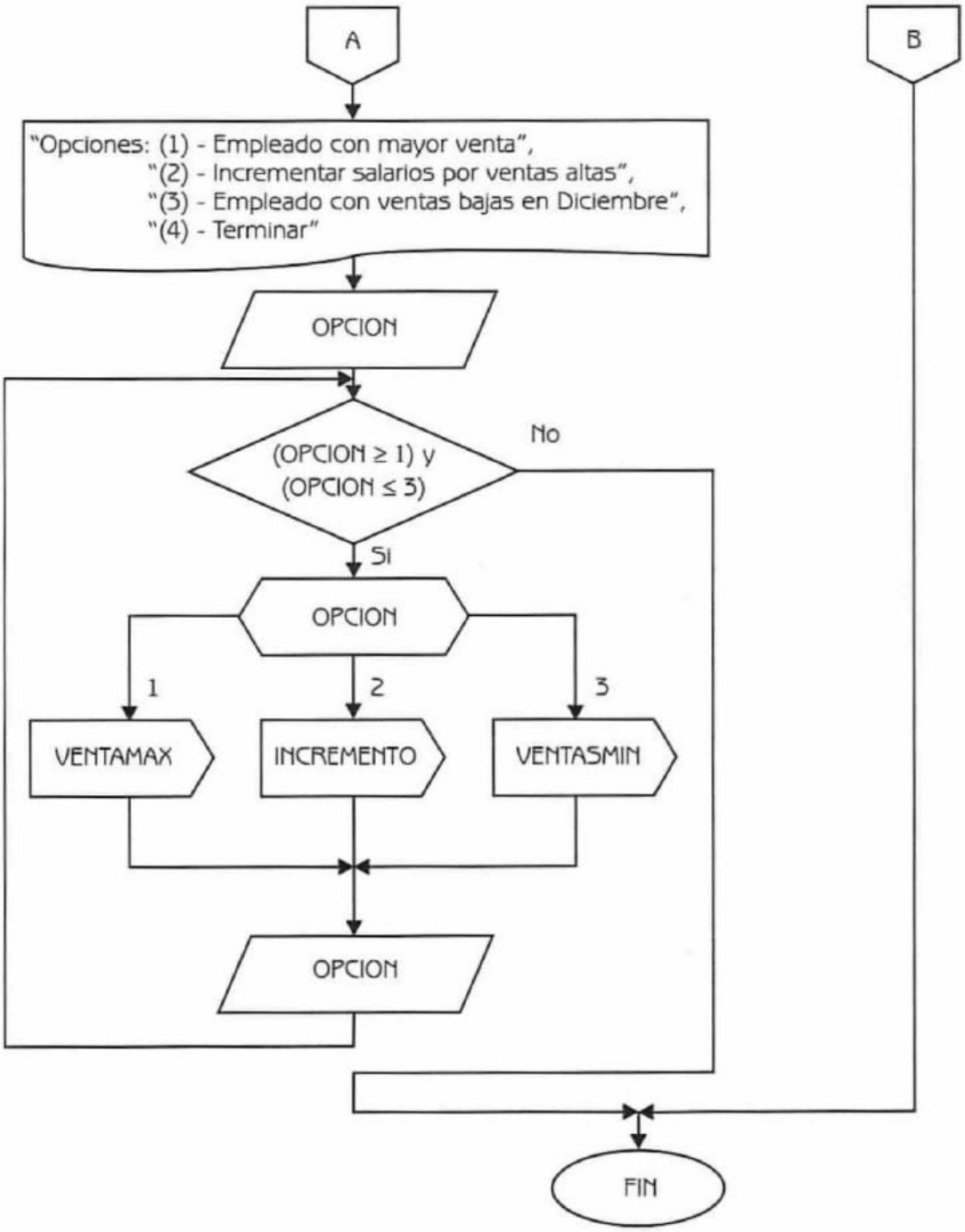


Diagrama de flujo 5.2 (*continuación...*)

www.FreeLibros.com

Nota:
Cálculo del inciso a.

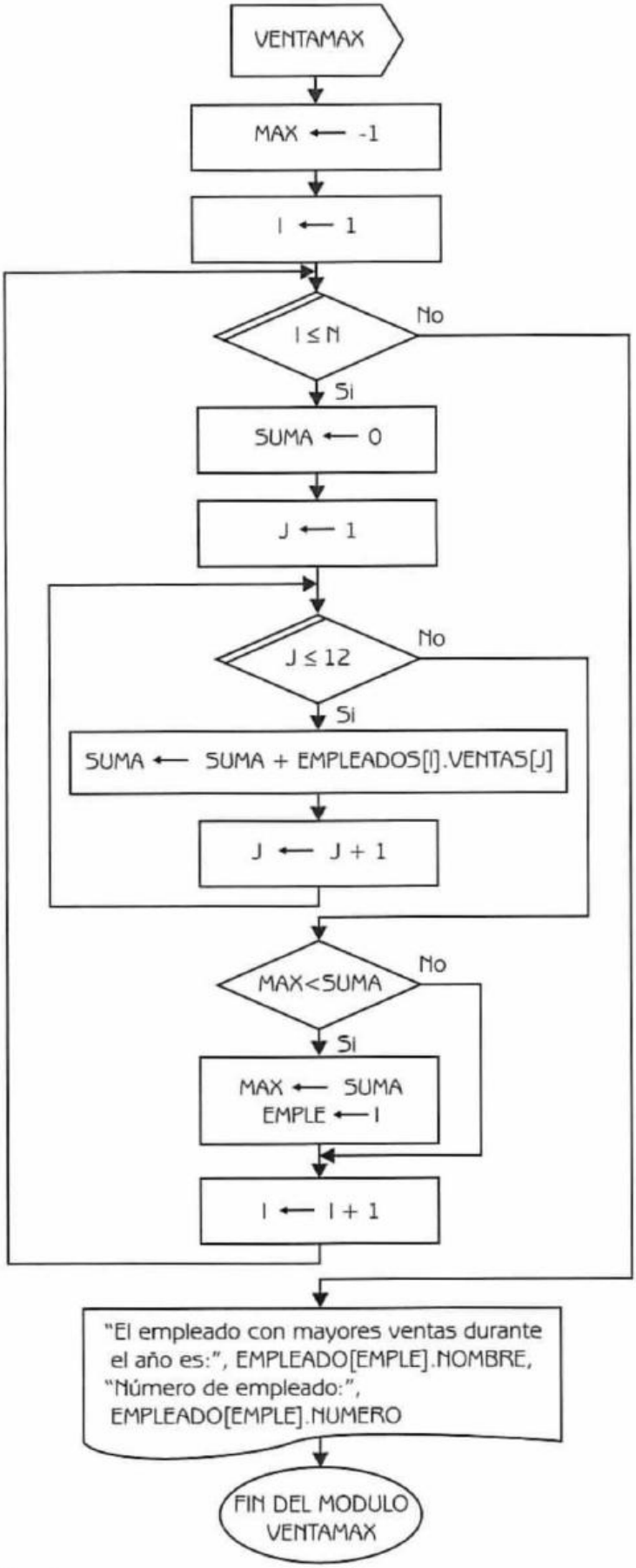


Diagrama de Flujo 5.2 *(continuación...)*

www.FreeLibros.com

Nota:
Cálculo del inciso b.

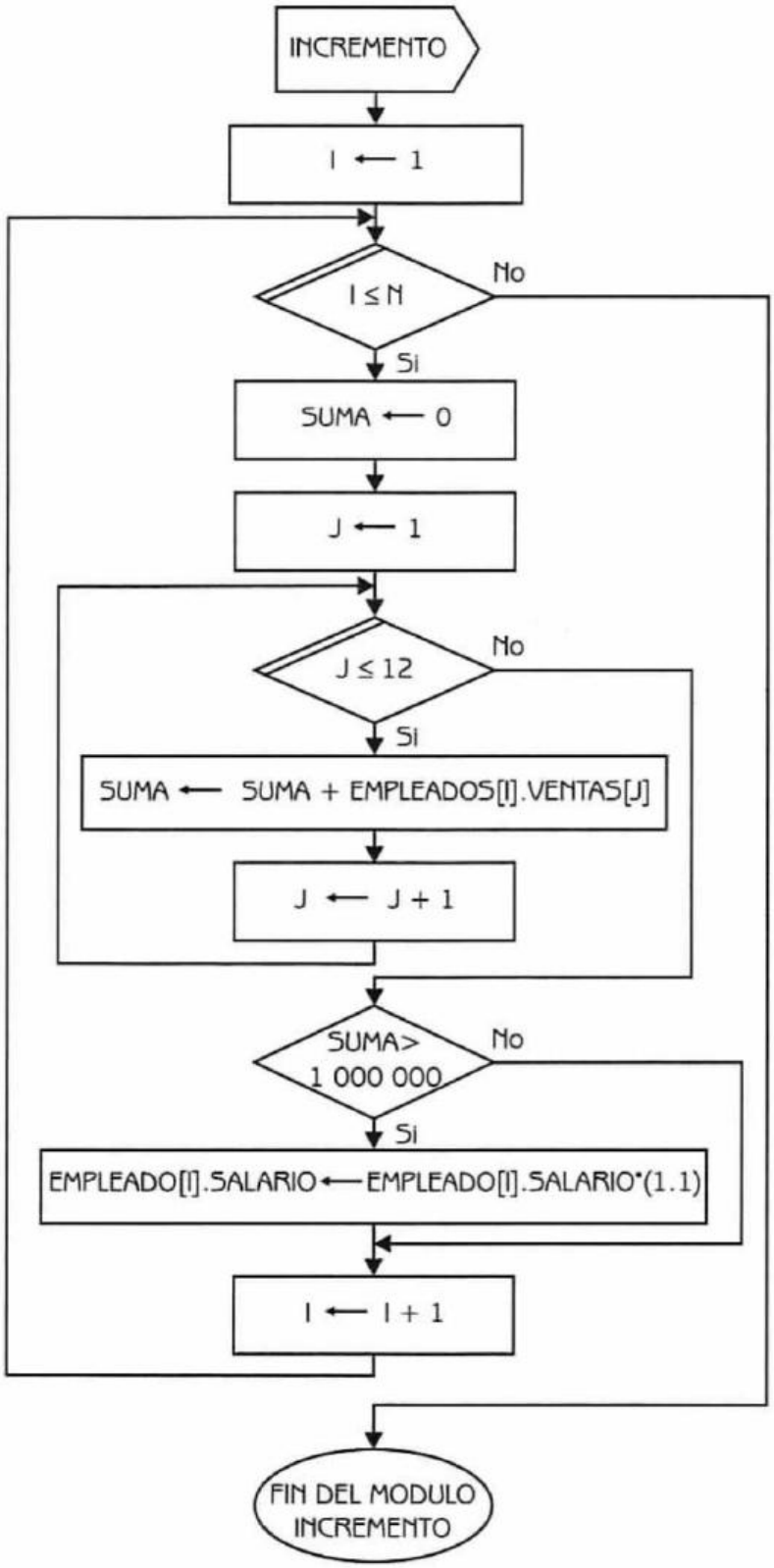
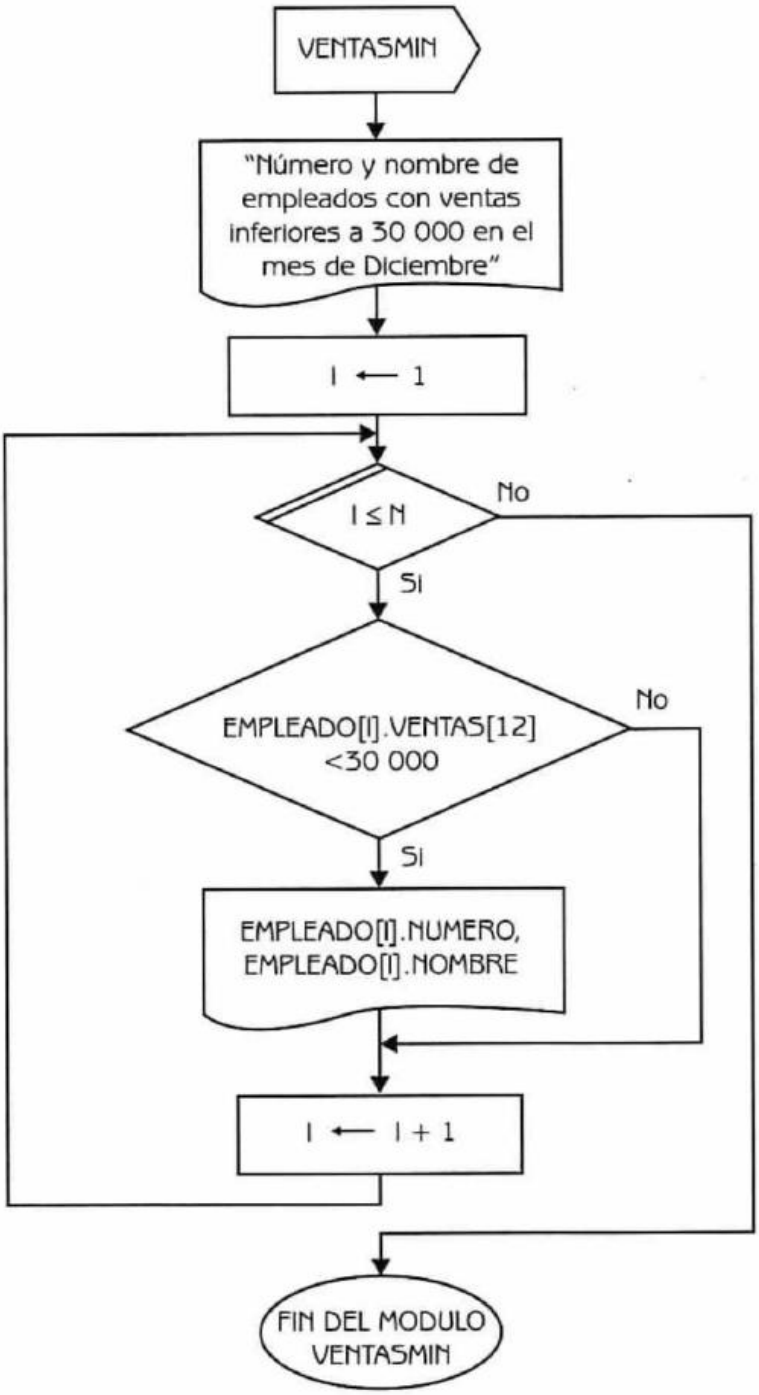


Diagrama de Flujo 5.2 (*continuación...*)

www.FreeLibros.com



Nota:
Cálculo del inciso c.

Diagrama de Flujo 5.2 (*continuación*)

www.FreeLibros.com

Explicación de las variables

N: Variable de tipo entero. Indica el total de empleados.

I: Variable de tipo entero. Se usa como índice del arreglo y como variable de control de ciclos.

EMPLEADOS: Arreglo unidimensional de registros.

J: Variable de tipo entero. Se utiliza como índice del arreglo " ventas" (que es un campo del registro), y como variable de control de ciclos.

OPCION: Variable de tipo entero. Se utiliza para leer la opción de trabajo seleccionada por el usuario.

MAX: Variable de tipo real. Se usa en la solución del inciso a, para obtener el número y nombre del empleado que realizó la mayor venta durante el año.

EMPLE: Variable de tipo entero. Almacena la posición en la que se encuentran los datos del empleado que realizó más ventas durante el año (inciso a).

SUMA: Variable de tipo real. Se usa para sumar las ventas realizadas por cada empleado a lo largo del año (incisos a y b).

Programa 5.2

```
EMPLEADOS_TIENDA

{El programa, dada información sobre las ventas efectuadas por los empleados de
una tienda en el año anterior, efectúa las siguientes operaciones: a) lista el núme-
ro y nombre del empleado que tuvo las mayores ventas durante el año, b) incre-
menta un 10% el salario de todos los empleados que durante el año obtuvieron
ventas superiores a $1 000 000, y c) lista el número y nombre de los empleados
que en el mes de diciembre tuvieron ventas inferiores a $30 000}

{N, I, J, OPCION y EMPLE son variables de tipo entero. MAX y SUMA son variables de
tipo real. EMPLEADOS es un arreglo unidimensional de registros}

1. Escribir " Ingresa número de empleados"
2. Leer N
3. Si (N >= 1) y (N <= 100)
        entonces
```

www.FreeLibros.com

```
      Hacer I ⟵ 1
3.1 Repetir con I desde 1 hasta N
          Escribir " Ingrese número, nombre y salario del empleado", I
          Leer EMPLEADOS[I].NUMERO, EMPLEADOS[I].NOMBRE y
               EMPLEADOS[I].SALARIO
          Hacer J ⟵ 1
  3.1.1 Repetir con J desde 1 hasta 12
          Escribir " Ingreso total de ventas en el mes", J,
                   " del empleado", EMPLEADOS[I].NOMBRE
          Leer EMPLEADOS[I].VENTAS[J]
          Hacer J ⟵ J+1
  3.1.2 {Fin del ciclo del paso 3.1.1}
          Hacer I ⟵ I+1
3.2 {Fin del ciclo del paso 3.1}
      Escribir " Opciones:  (1) - Empleado con mayor venta",
                          " (2) - Incrementar salario por ventas altas",
                          " (3) - Empleados con ventas bajas en Diciembre",
                          " (4) - Terminar"
      Leer OPCION
3.3 Mientras (OPCION >= 1) y (OPCION <= 3) repetir
  3.3.1 Si OPCION igual
          1 : Hacer MAX ⟵ -1 e I ⟵ 1
    3.3.1.1 Repetir con I desde 1 hasta N
                  Hacer SUMA ⟵ 0 y J ⟵ 1
      3.3.1.1.1 Repetir con J desde 1 hasta 12
                      Hacer SUMA ⟵ SUMA +
                           EMPLEADOS[I].VENTAS[J]
                           y J ⟵ J+1
      3.3.1.1.2 {Fin del ciclo del paso 3.3.1.1.1}
      3.3.1.1.3 Si MAX < SUMA entonces
                      Hacer MAX ⟵ SUMA y EMPLE ⟵ I
      3.3.1.1.4 {Fin del condicional del paso 3.3.1.1.3}
                  Hacer I ⟵ I+1
    3.3.1.2 {Fin del ciclo del paso 3.3.1.1}
              Escribir " El empleado con mayores ventas durante el año es:",
                       EMPLEADO[EMPLE].NOMBRE,
                       " Número de empleado:", EMPLEADO[EMPLE].NUMERO
          2: Hacer I ⟵ 1
    3.3.1.3 Repetir con I desde 1 hasta N
                  Hacer SUMA ⟵ 0 y J ⟵ 1
      3.3.1.3.1 Repetir con J desde 1 hasta 12
                      Hacer SUMA ⟵ SUMA +
                           EMPLEADOS[I].VENTAS[J] y J ⟵ J+1
```

www.FreeLibros.com

```
            3.3.1.3.2 {Fin del ciclo del paso 3.3.1.3.1}
            3.3.1.3.3 Si SUMA > 1 000 000 entonces
                          Hacer EMPLEADOS[I]. SALARIO ←——
                                EMPLEADOS[I].SALARIO * 1.1
            3.3.1.3.4 {Fin del condicional del paso 3.3.1.3.3}
                      Hacer I ←—— I+1
          3.3.1.4 {Fin del ciclo del paso 3.3.1.3}
                3: Escribir " Número y nombre de empleados con ventas
                             inferiores a 30 000 en el mes de Diciembre"
                   Hacer I ←—— 1
          3.3.1.5 Repetir con I desde 1 hasta N
            3.3.1.5.1 Si EMPLEADOS[I].VENTAS[12] < 30 000 entonces
                              Escribir EMPLEADOS[I].NUMERO y
                                       EMPLEADOS[I].NOMBRE
            3.3.1.5.2 {Fin del condicional del paso 3.3.1.5.1}
                      Hacer I ←—— I+1
          3.3.1.6 {Fin del ciclo del paso 3.3.1.5}
        3.3.2 {Fin del condicional del paso 3.3.1}
              Leer OPCION
    3.4 {Fin del ciclo del paso 3.3}
    sino
          Escribir " Error en los datos"
4. {Fin del condicional del paso 3}
```

Problema 5.3

En una universidad almacenan información relativa a sus profesores en un arreglo unidimensional. Cada elemento del arreglo es un registro que contiene la siguiente información respecto a cada profesor:

- Número de empleado.
- Nombre del profesor.
- Departamento al que pertenece.
- Grado académico.
- Nacionalidad.
- Salario (Se almacena en forma mensual lo que cobra el profesor en un arreglo unidimensional).

Construya un diagrama de flujo que pueda proporcionar la siguiente información:

www.FreeLibros.com

a) El número, nombre y nacionalidad del profesor que más ganó el año anterior.

b) El monto total pagado en el año a los profesores del departamento X.

c) El monto total pagado a los profesores extranjeros (nacionalidad distinta a México).

d) El nombre del profesor del departamento Y que más ganó en el año anterior.

Dato: PROFESORES [1..N] $1 \leq N \leq 200$

Donde:

PROFESORES es un arreglo unidimensional de registros.

Cada registro a su vez tiene los siguientes campos:

NUM: Variable de tipo entero que representa el número del empleado.

NOM: Variable de tipo cadena de caracteres que indica el nombre del profesor.

DEP: Variable de tipo cadena de caracteres que representa el departamento en que está adscrito el profesor.

GRADO: Variable de tipo cadena de caracteres que expresa el máximo grado académico del profesor.

NAC: Variable de tipo cadena de caracteres que representa la nacionalidad del profesor.

SALARIO: Arreglo unidimensional de tipo real que almacena en forma mensual lo que cobró el profesor.

www.FreeLibros.com

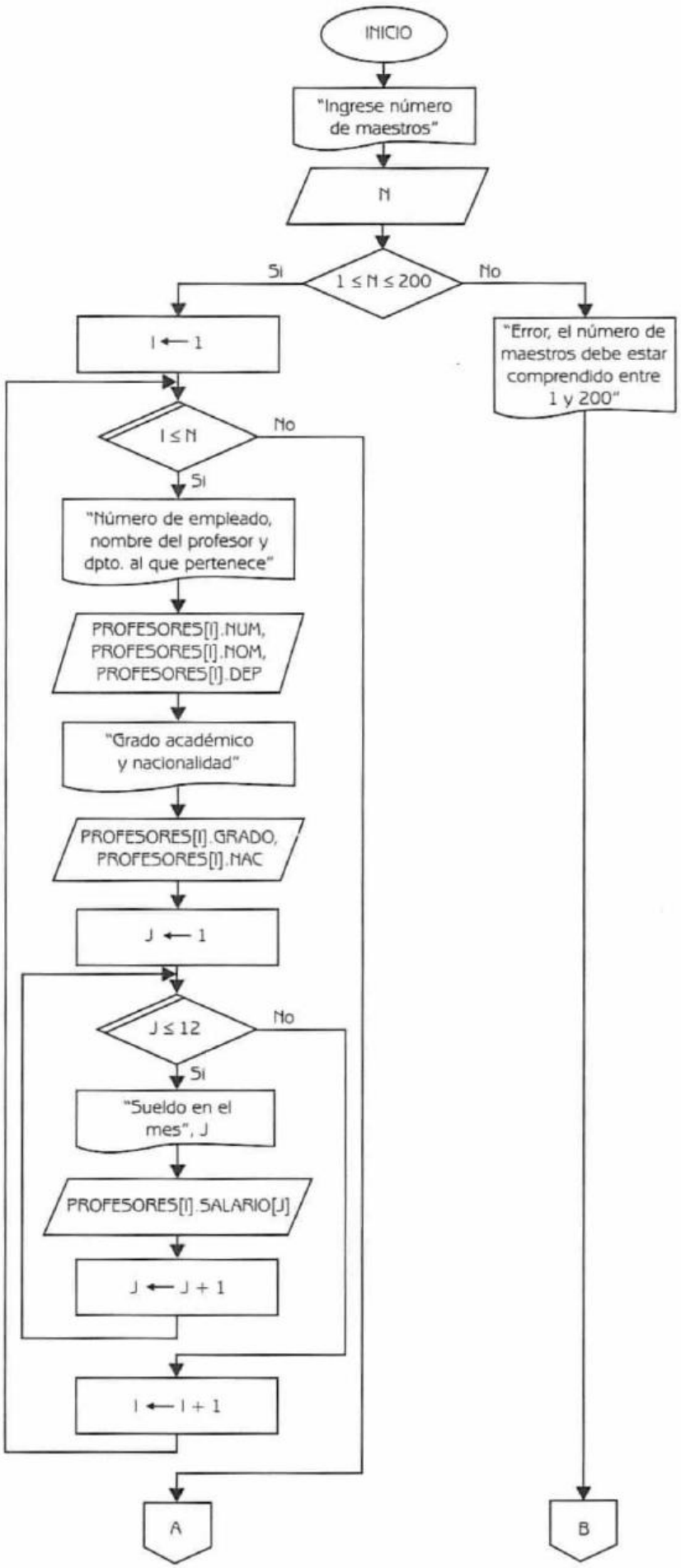


Diagrama de Flujo 5.3 (*continúa*)

www.FreeLibros.com

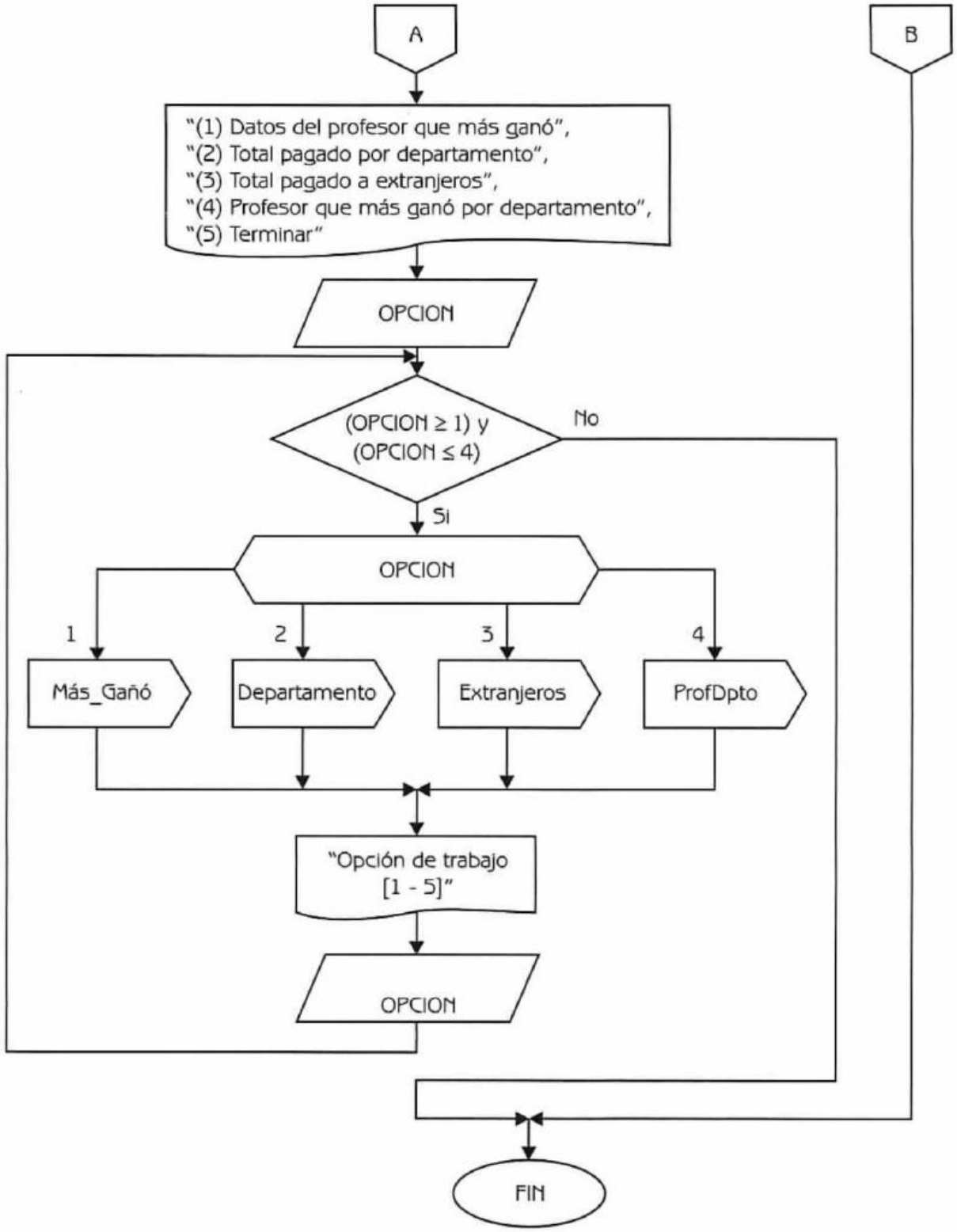


Diagrama de Flujo 5.3 (*continuación...*)

www.FreeLibros.com

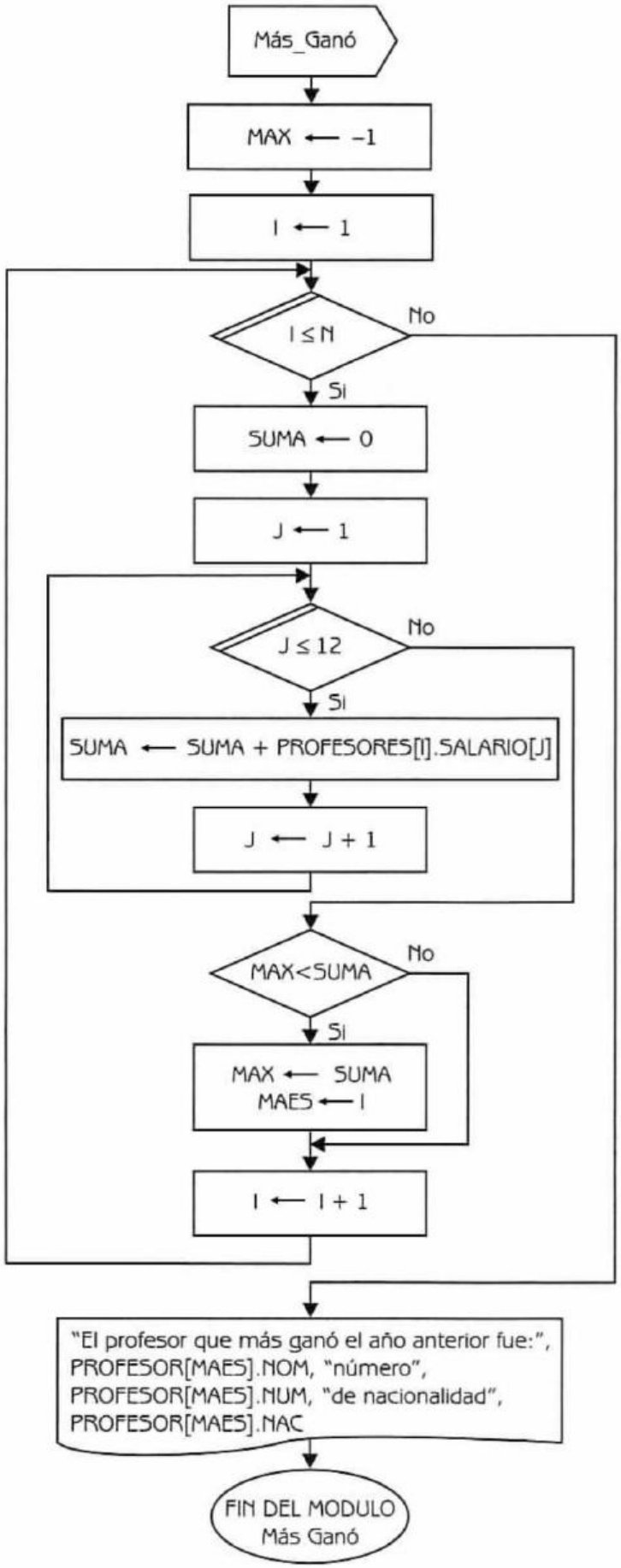


Diagrama de Flujo 5.3 (*continuación...*)

www.FreeLibros.com

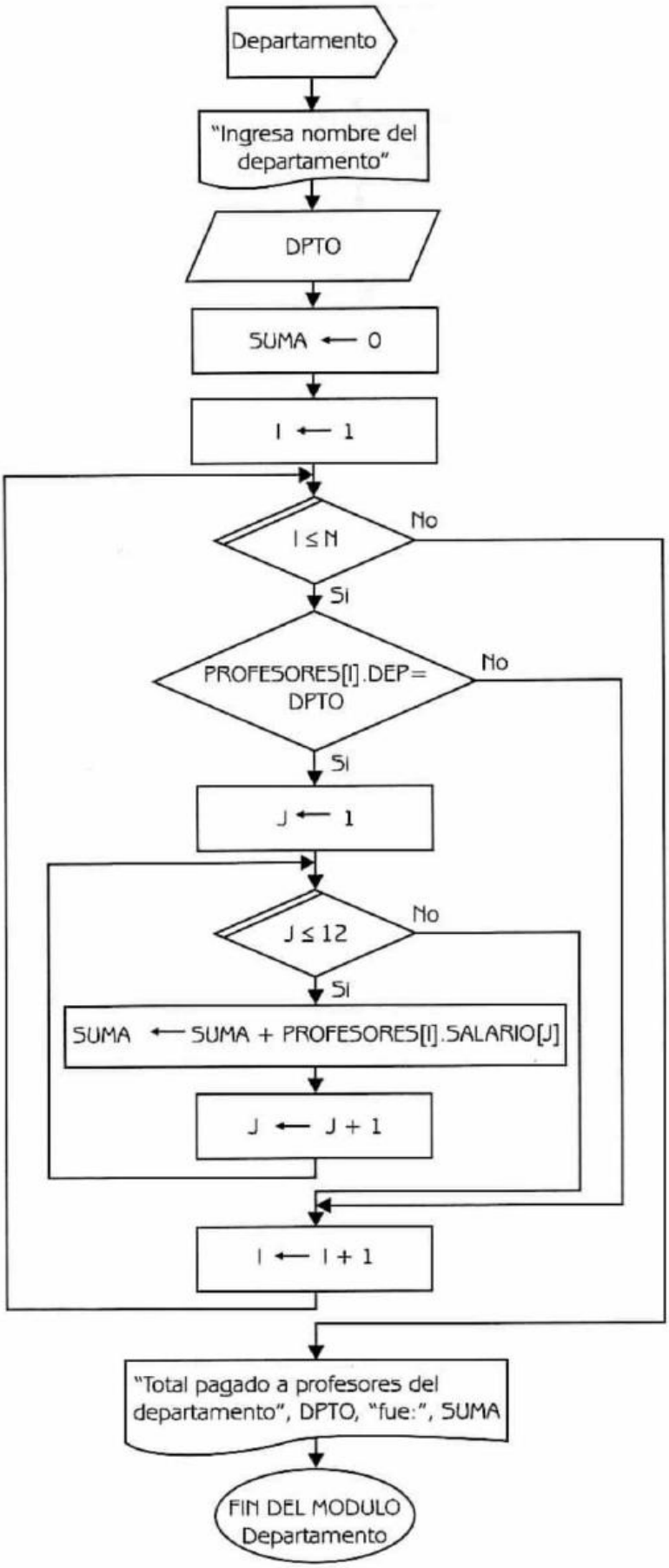


Diagrama de Flujo 5.3 (*continuación...*)

www.FreeLibros.com

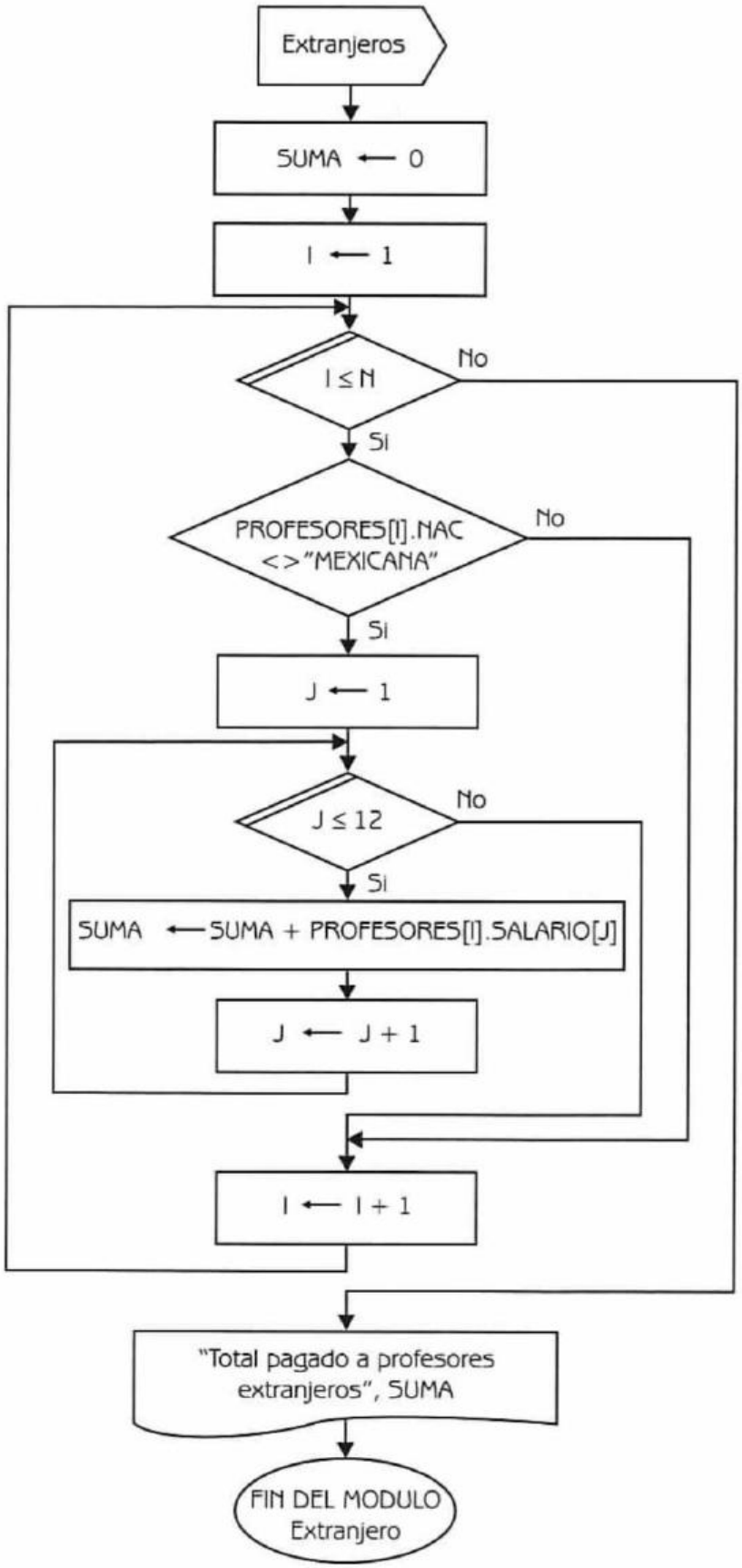


Diagrama de Flujo 5.3 (*continuación...*)

www.FreeLibros.com

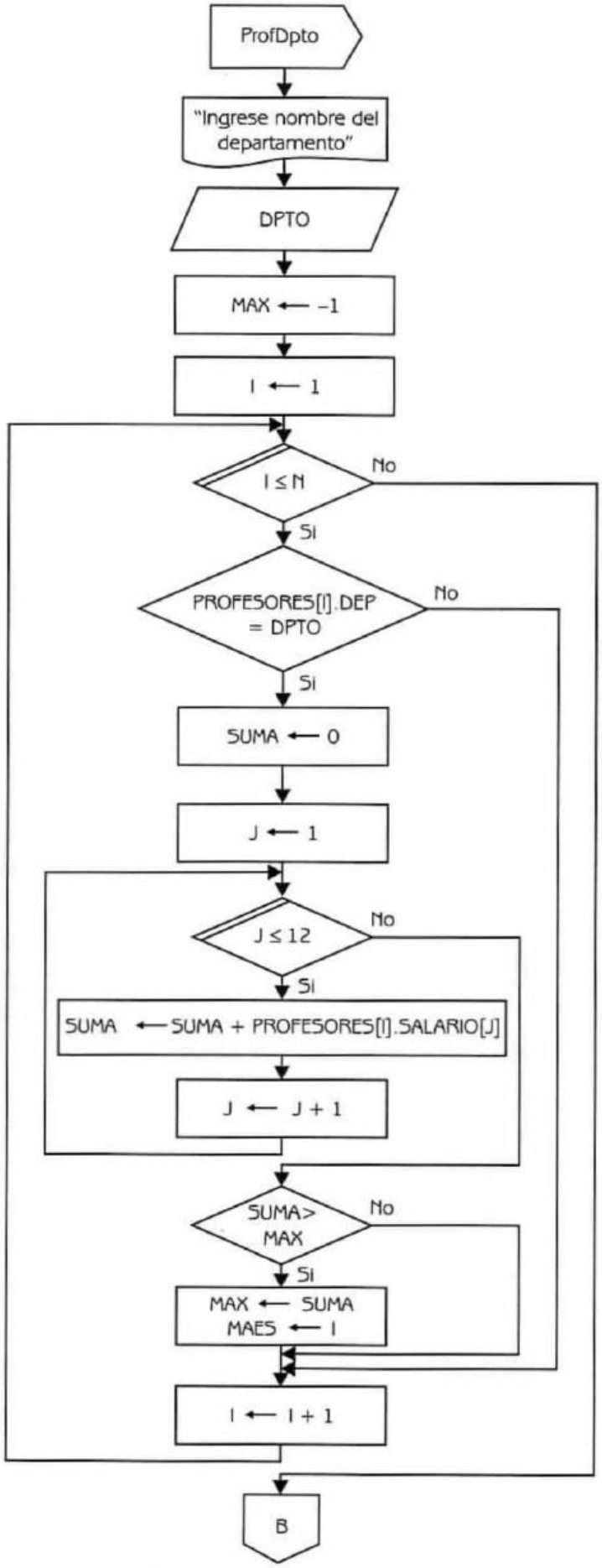


Diagrama de Flujo 5.3 *(continuación...)*

www.FreeLibros.com

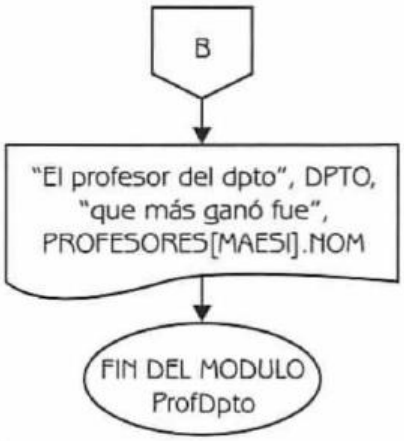


Diagrama de Flujo 5.3 (*continuación*)

Explicación de las variables

N: Variable de tipo entero. Indica el número de maestros.

I: Variable de tipo entero. Se usa como variable de control del ciclo y como índice del arreglo.

J: Variable de tipo entero. Se usa como variable de control del ciclo y como índice del arreglo.

PROFESORES: Arreglo unidimensional de tipo registro.

OPCION: Variable de tipo entero. Se usa para leer la opción de trabajo seleccionada por el usuario.

MAX: Variable de tipo real. Se utiliza para almacenar la cantidad máxima ganada por un profesor (incisos a y d).

SUMA: Variable de tipo real. Se usa para sumar salarios recibidos por los maestros a lo largo del año.

MAES: Variable de tipo entero. Almacena la posición en la que se encuentran registrados los datos del maestro que más ganó (incisos a y d).

DPTO: Variable de tipo cadena de caracteres. Se usa para leer el nombre de un departamento (inciso b y d).

www.FreeLibros.com

Programa 5.3

```
PROFESORES_UNIVERSIDAD

{El programa, dada información relativa a los profesores de una universidad, obtiene
la siguiente información: a) El número, nombre y nacionalidad del profesor que más
ganó el año anterior, b) El monto total pagado en el año a los profesores del departa-
mento X, c) El monto total pagado a los profesores extranjeros, y d) El nombre del
profesor del departamento que más ganó en el año anterior}

{N, I, J, OPCION y MAES son variables de tipo entero. MAX y SUMA son variables de
tipo real. DPTO es una variable de tipo cadena de caracteres. PROFESORES es un
arreglo unidimensional de tipo registro}

1. Escribir "Ingrese número de maestros"
2. Leer N
3. Si (N >= 1) y (N <= 200)
        entonces
          Hacer I ←—— 1
    3.1 Repetir con I desde 1 hasta N
        Escribir "Número de empleado, nombre del profesor y dpto. al que pertenece"
          Leer PROFESORES[I].NUM, PROFESORES[I].NOM y
              PROFESORES[I].DEP
        Escribir "Grado académico y nacionalidad"
        Leer PROFESORES[I].GRADO y PROFESORES[I].NAC
        Hacer J ←—— 1
      3.1.1 Repetir con J desde 1 hasta 12
                Escribir "Sueldo en el mes", J
                Leer PROFESORES[I].SALARIO[J]
                Hacer J ←—— J + 1
      3.1.2 {Fin del ciclo del paso 3.1.1}
              Hacer I ←—— I + 1
    3.2 {Fin del ciclo del paso 3.1}
        Escribir "(1) - Datos del Profesor que más ganó",
                 "(2) - Total pagado por departamento",
                 "(3) - Total pagado a extranjeros",
                 "(4) - Profesor que más ganó por departamento",
                 "(5) - Terminar"
        Leer OPCION
    3.3 Mientras (OPCION >= 1) y (OPCION <= 4) repetir
      3.3.1 Si OPCION igual
              1: Hacer MAX ←—— -1 e I ←—— 1
```

www.FreeLibros.com

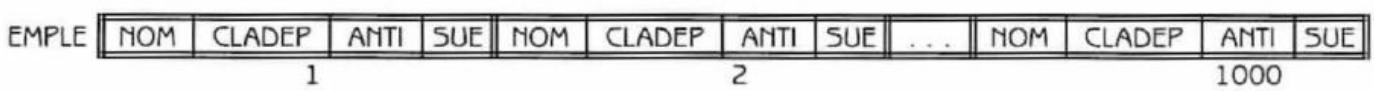
```
3.3.1.1 Repetir con I desde 1 hasta N
            Hacer SUMA ⟵ 0 y J ⟵ 1
  3.3.1.1.1 Repetir con J desde 1 hasta 12
                Hacer SUMA ⟵ SUMA
                    + PROFESORES[I].SALARIO[J] y J ⟵ J + 1
  3.3.1.1.2 {Fin del ciclo del paso 3.3.1.1.1}
  3.3.1.1.3 Si MAX < SUMA entonces
                Hacer MAX ⟵ SUMA y MAES ⟵ I
  3.3.1.1.4 {Fin del condicional del paso 3.3.1.1.3}
            Hacer I ⟵ I + 1
3.3.1.2 {Fin del ciclo del paso 3.3.1.1}
        Escribir "El profesor que más ganó en el año anterior fue:",
                 PROFESOR[MAES].NOM, "número:",
                 PROFESOR[MAES].NUM, "de nacionalidad:",
                 PROFESOR[MAES].NAC
      2: Escribir "Ingrese nombre del departamento"
         Leer DPTO
         Hacer SUMA ⟵ 0 e I ⟵ 1
3.3.1.3 Repetir con I desde 1 hasta N
  3.3.1.3.1 Si PROFESORES[I].DEP = DPTO entonces
                Hacer J ⟵ 1
    3.3.1.3.1.1 Repetir con J desde 1 hasta 12
                    Hacer SUMA ⟵ SUMA
                        + PROFESORES[I].SALARIO[J]
                        y J ⟵ J + 1
    3.3.1.3.1.2 {Fin del ciclo del paso 3.3.1.3.1.1}
  3.3.1.3.2 {Fin del condicional del paso 3.3.1.3.1}
            Hacer I ⟵ I + 1
3.3.1.4 {Fin del ciclo del paso 3.3.1.3}
        Escribir "Total pagado a profesores del departamento", DPTO,
                 "fue:", SUMA
      3: Hacer SUMA ⟵ 0 e I ⟵ 1
3.3.1.5 Repetir con I desde 1 hasta N
  3.3.1.5.1 Si PROFESORES[I].NAC <> "MEXICANA" entonces
            Hacer J ⟵ 1
    3.3.1.5.1.1 Repetir con J desde 1 hasta 12
                    Hacer SUMA ⟵ SUMA
                        + PROFESORES[I].SALARIO[J]
                        y J ⟵ J + 1
    3.3.1.5.1.2 {Fin del ciclo del paso 3.3.1.5.1.1}
  3.3.1.5.2 {Fin del condicional del paso 3.3.1.5.1}
            Hacer I ⟵ I + 1
```

www.FreeLibros.com

```
        3.3.1.6 {Fin del ciclo del paso 3.3.1.5}
                Escribir "Total pagado a profesores extranjeros:", SUMA
           4: Escribir "Ingrese nombre del departamento"
              Leer DPTO
              Hacer MAX ⟵ -1 e I ⟵ 1
     3.3.1.7 Repetir con I desde 1 hasta N
       3.3.1.7.1 Si PROFESORES[I].DEP = DPTO entonces
                       Hacer SUMA ⟵ 0 y J ⟵ 1
         3.3.1.7.1.1 Repetir con J desde 1 hasta 12
                          Hacer SUMA ⟵ SUMA
                                + PROFESORES[I].SALARIO[J]
                                y J ⟵ J + 1
         3.3.1.7.1.2 {Fin del ciclo del paso 3.3.1.7.1.1}
         3.3.1.7.1.3 Si SUMA > MAX entonces
                          Hacer MAX ⟵ SUMA y MAES ⟵ I
         3.3.1.7.1.4 {Fin del condicional del paso 3.3.1.7.1.3}
       3.3.1.7.2 {Fin del condicional del paso 3.3.1.7.1}
                 Hacer I ⟵ I + 1
     3.3.1.8 {Fin del ciclo del paso 3.3.1.7}
             Escribir "El profesor del departamento", DPTO, "que más ganó
                      fue", PROFESORES[MAES].NOM
   3.3.2 {Fin del condicional del paso 3.3.1}
         Escribir "Opción de trabajo [1 - 5]"
         Leer OPCION
  3.4 {Fin del ciclo del paso 3.3}
  sino
     Escribir "Error, el número de maestros debe estar comprendido entre 1 y 200"
4. {Fin del condicional del paso 3}
```

Problema 5.4

Se tienen dos arreglos de registros que contienen la siguiente información:



Donde:

EMPLE[I] representa al registro que contiene información del empleado i. Los campos del registro son los siguientes:

www.FreeLibros.com

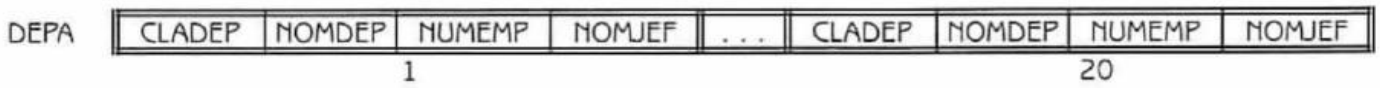
NOM: Expresa el nombre del empleado. Variable de tipo cadena de caracteres.

CLADEP: Expresa la clave del departamento donde trabaja el empleado. Variable de tipo entero.

ANTI: Expresa la antigüedad del empleado en años. Variable de tipo entero.

SUE: Representa el sueldo del empleado. Variable de tipo real.

Nota: El arreglo está ordenado teniendo en cuenta el nombre (NOM) del empleado.



Donde:

DEPA[I] representa al registro que contiene información del departamento i. Los campos del registro son los siguientes:

CLADEP: Expresa la clave del departamento. Variable de tipo entero.

NOMDEP: Representa el nombre del departamento. Variable de tipo cadena de caracteres.

NUMEMP: Expresa el número de empleados que tiene el departamento. Variable de tipo entero.

NOMJEF: Representa el nombre del jefe del departamento. Variable de tipo cadena de caracteres.

Nota: El arreglo está ordenado teniendo en cuenta la clave del departamento.

Construya un diagrama de flujo, que luego de leer los arreglos, pueda efectuar las siguientes operaciones:

a) Dado el nombre de un empleado, proporcione su sueldo, el nombre del departamento en que trabaja y el nombre de su jefe.
b) Dado el nombre de un departamento, obtenga sueldo y nombre del empleado que más gana en ese departamento.
c) Dado el nombre de un nuevo empleado, insértelo de forma correcta en los arreglos correspondientes.

www.FreeLibros.com

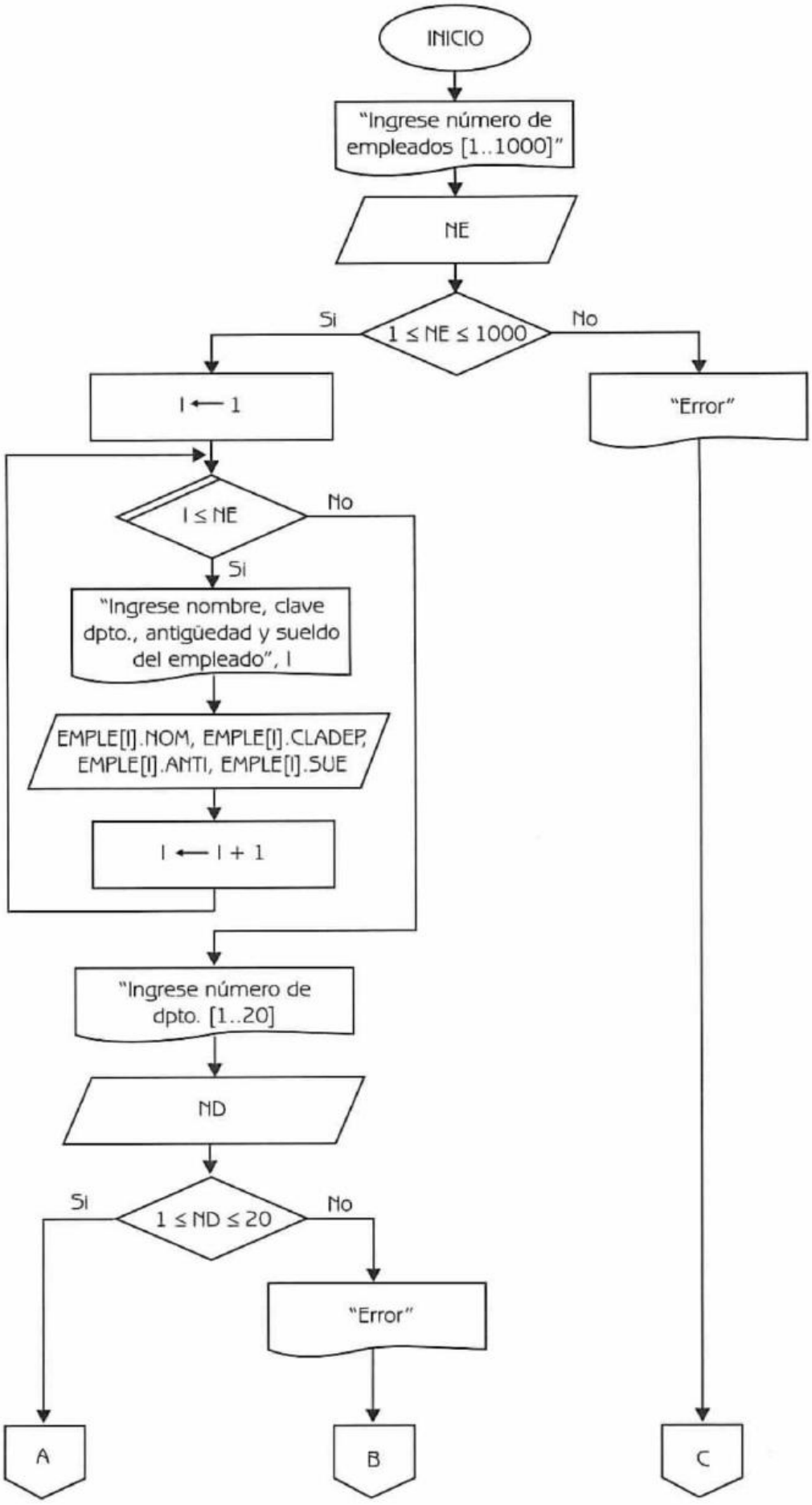


Diagrama de Flujo 5.4 *(continúa)*

www.FreeLibros.com

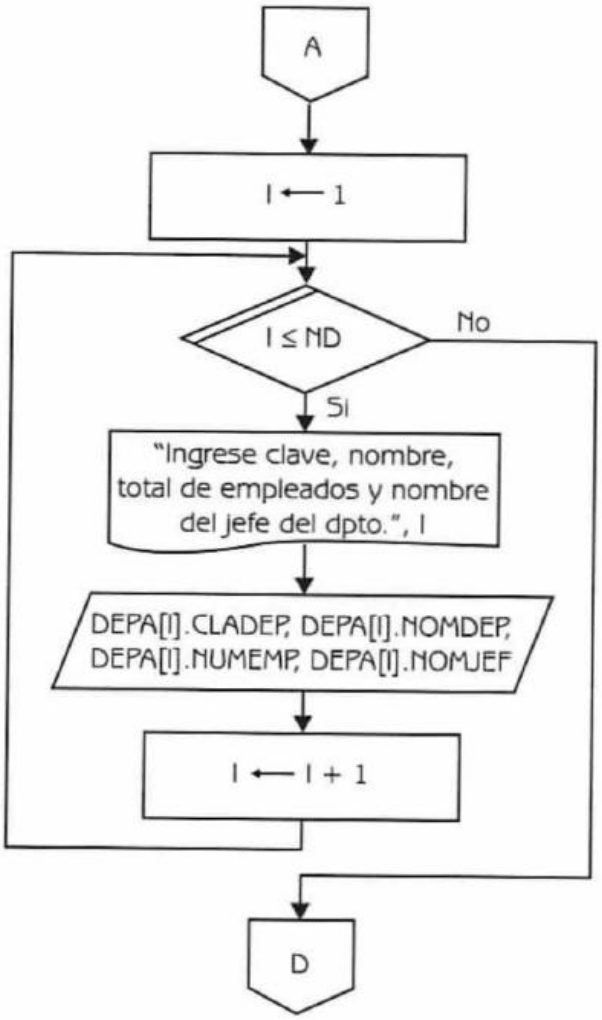


Diagrama de Flujo 5.4 *(continuación...)*

www.FreeLibros.com

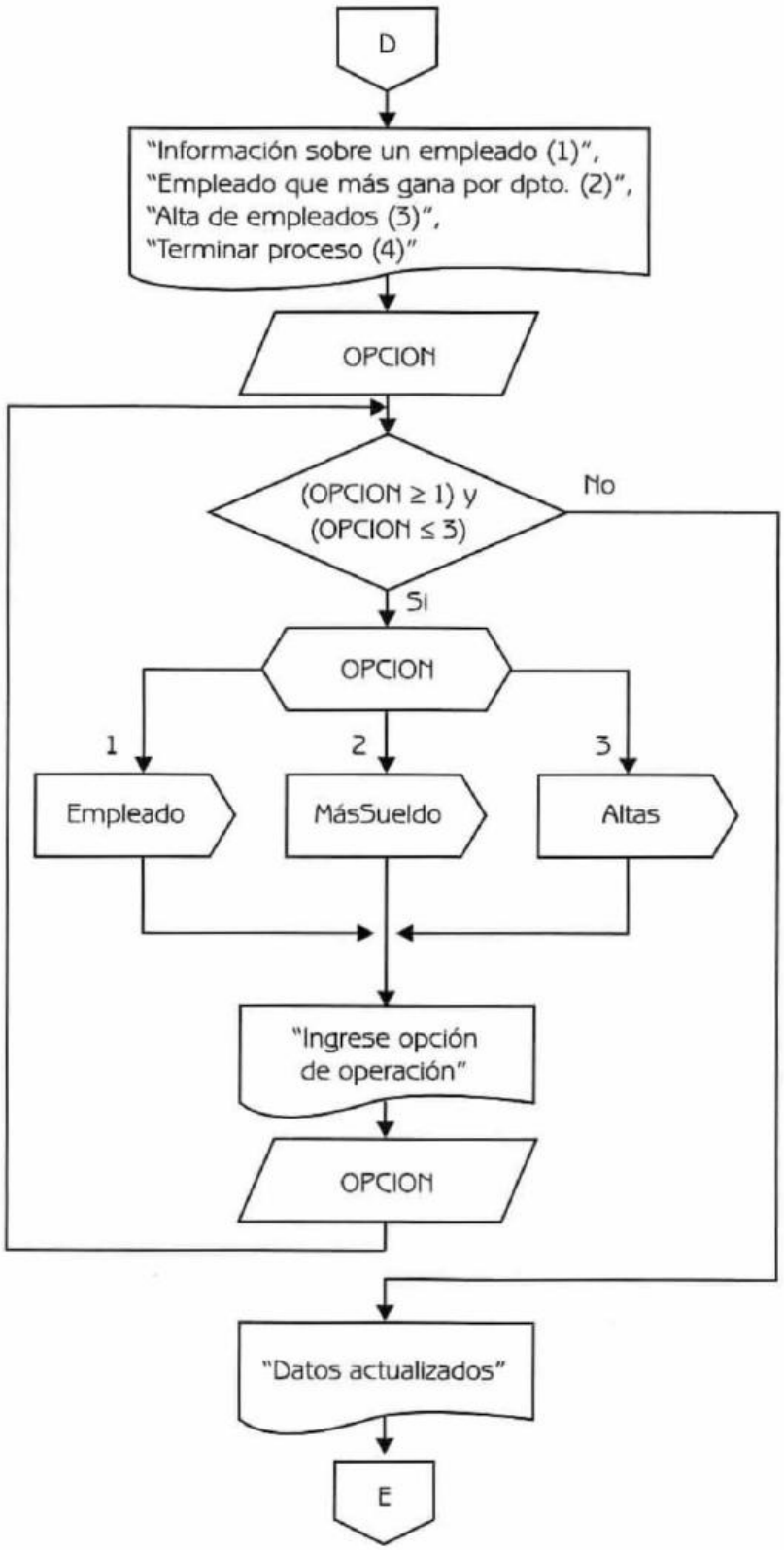


Diagrama de Flujo 5.4 (*continuación...*)

www.FreeLibros.com

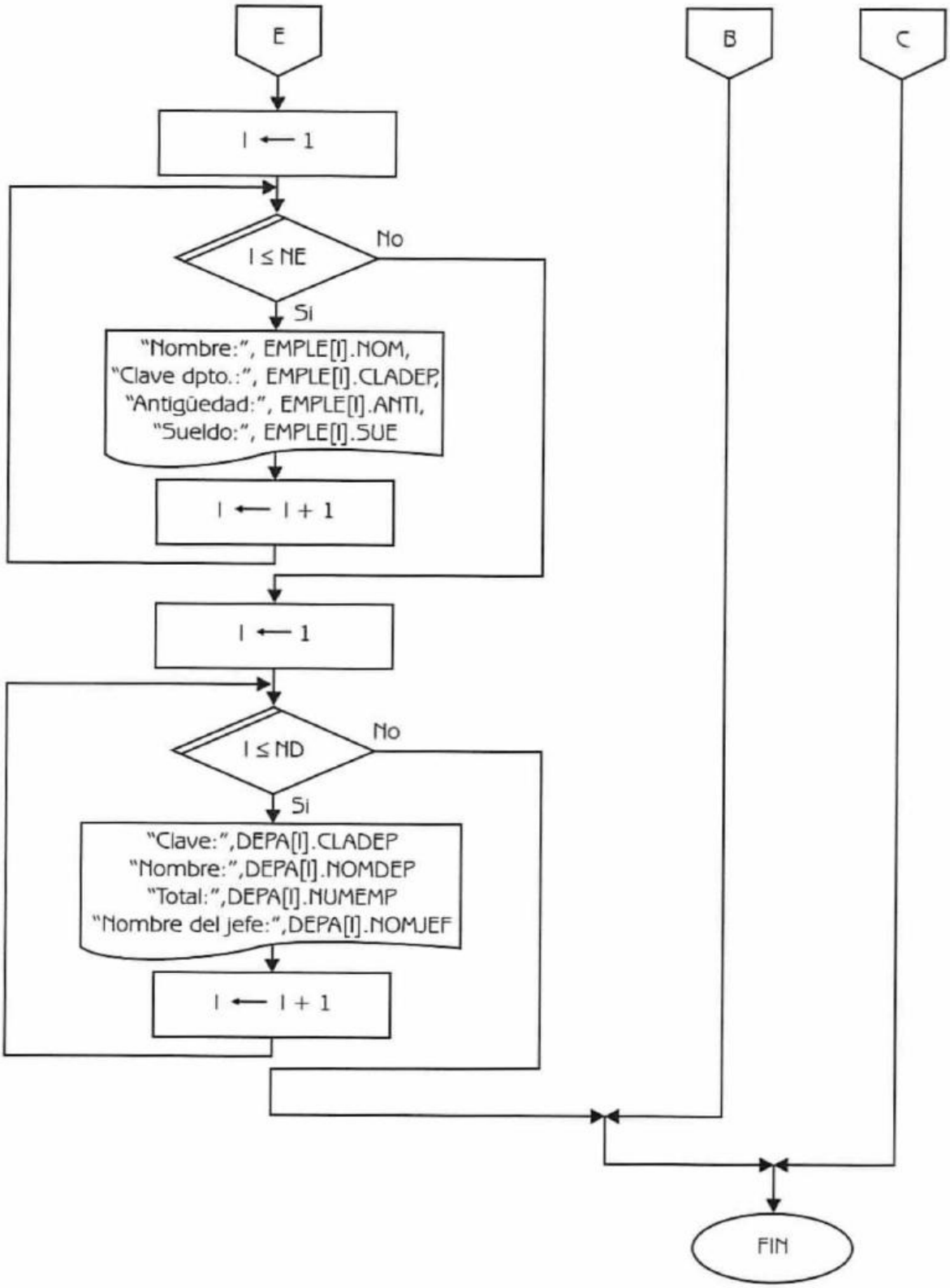


Diagrama de Flujo 5.4 (*continuación...*)

www.FreeLibros.com

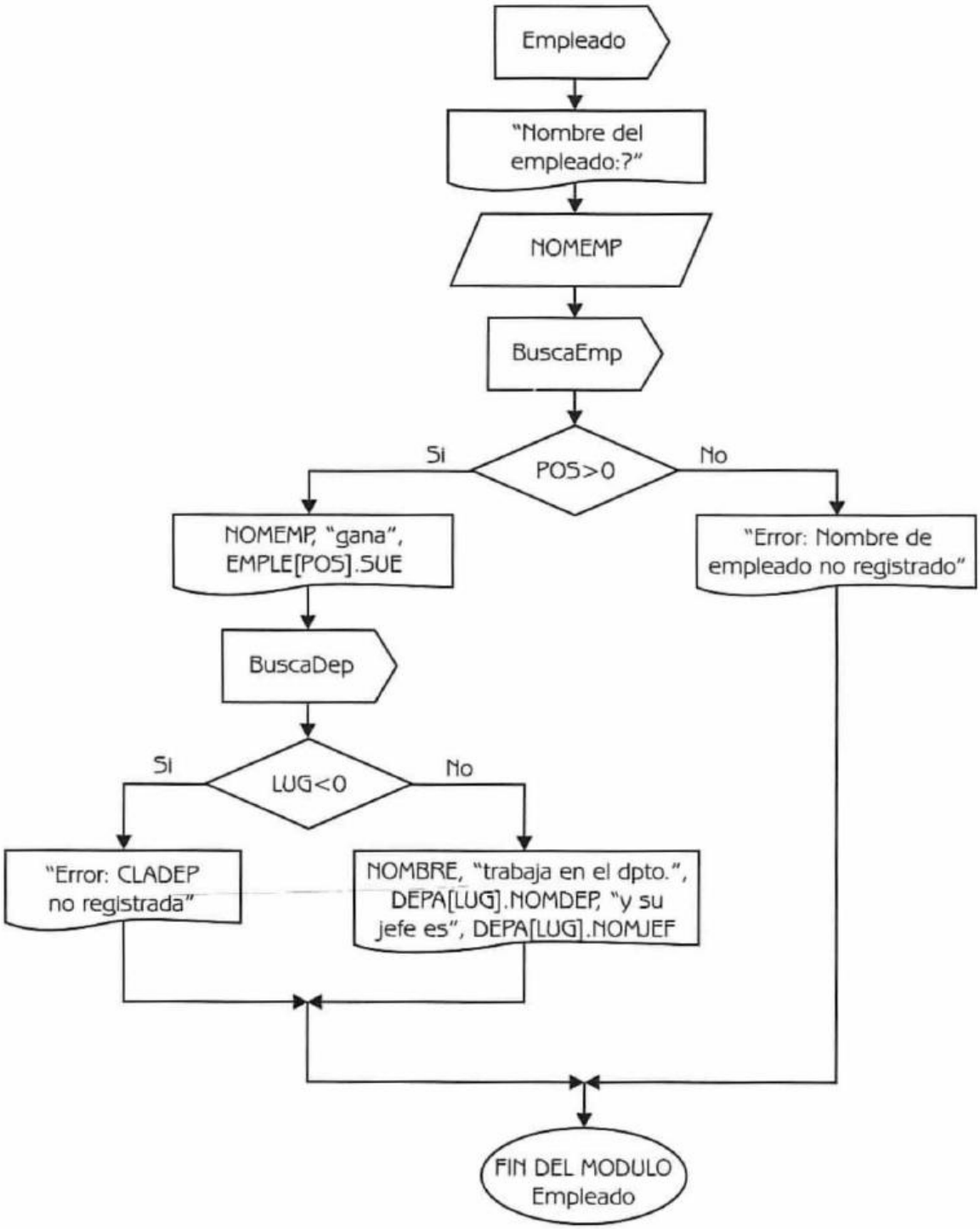


Diagrama de Flujo 5.4 (*continuación...*)

www.FreeLibros.com

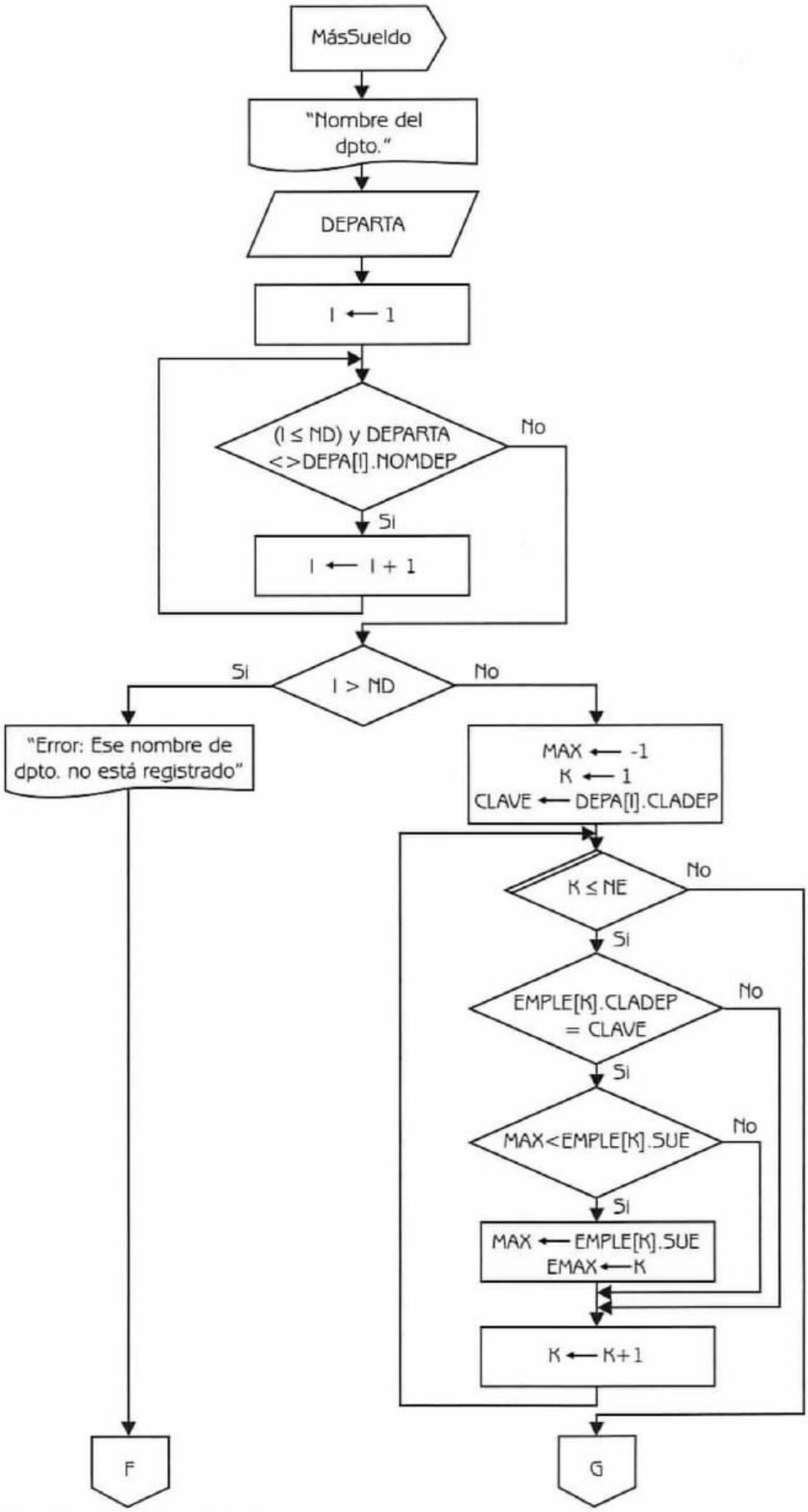


Diagrama de Flujo 5.4 (*continuación...*)

www.FreeLibros.com

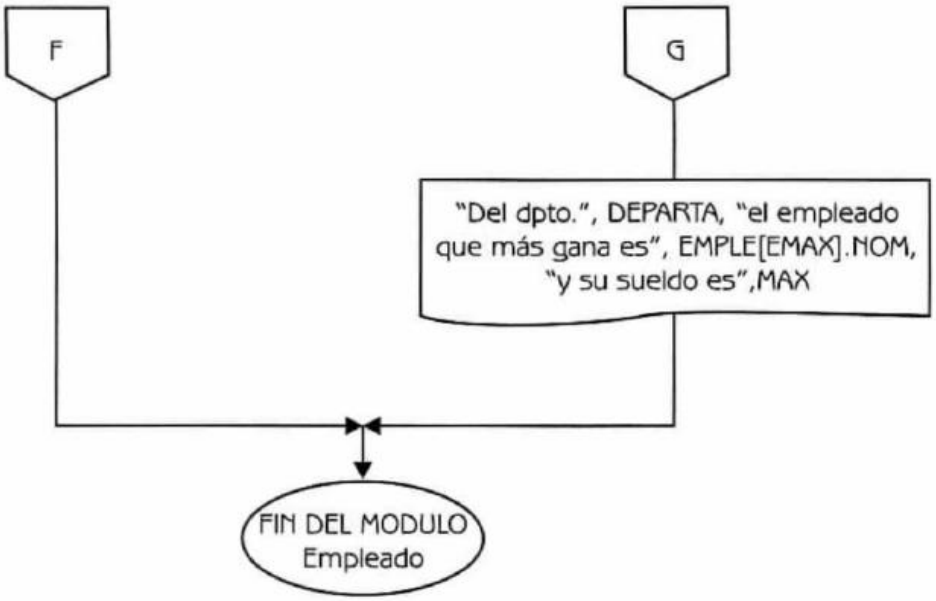


Diagrama de Flujo 5.4 (*continuación...*)

www.FreeLibros.com

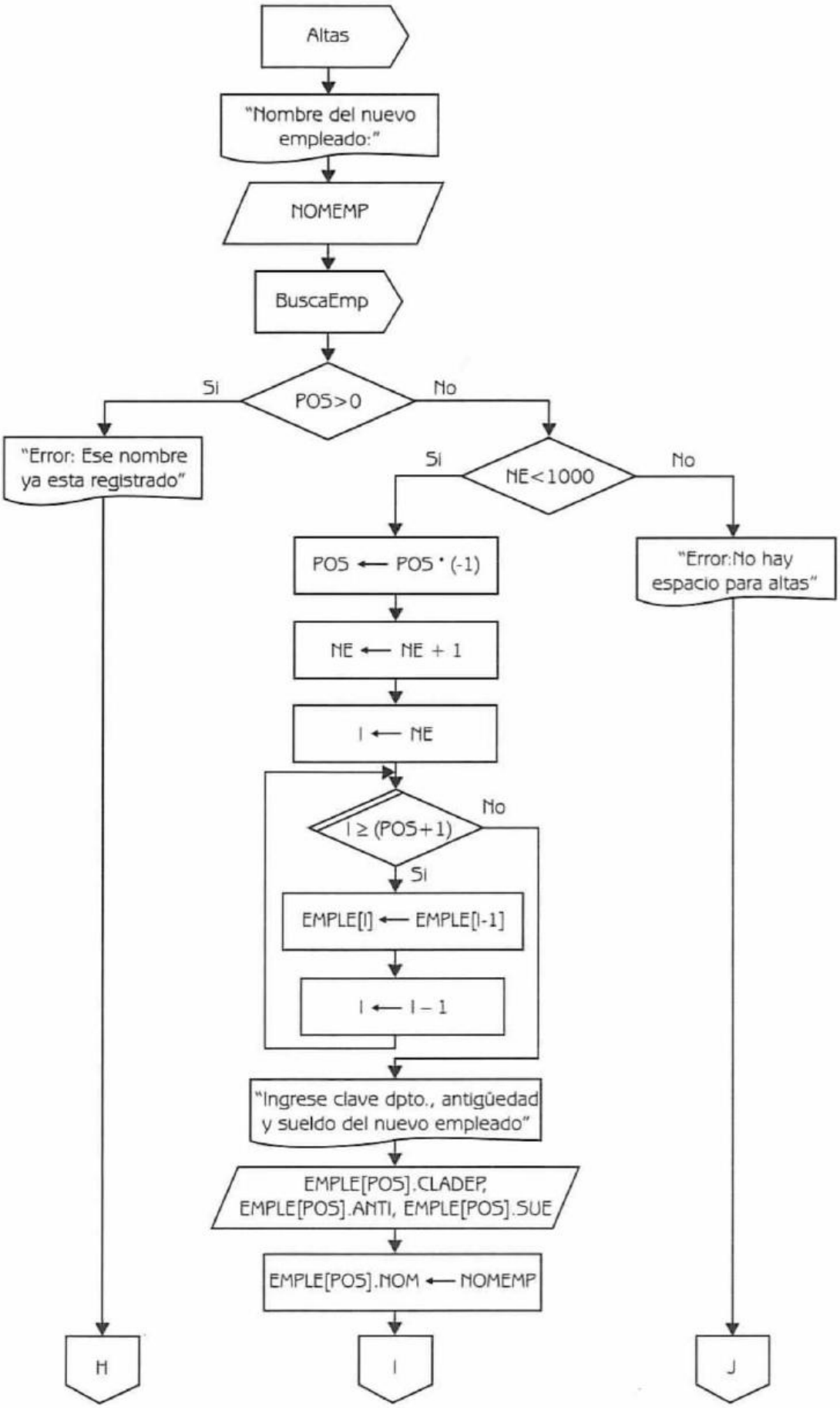


Diagrama de Flujo 5.4 (*continuación...*)

www.FreeLibros.com

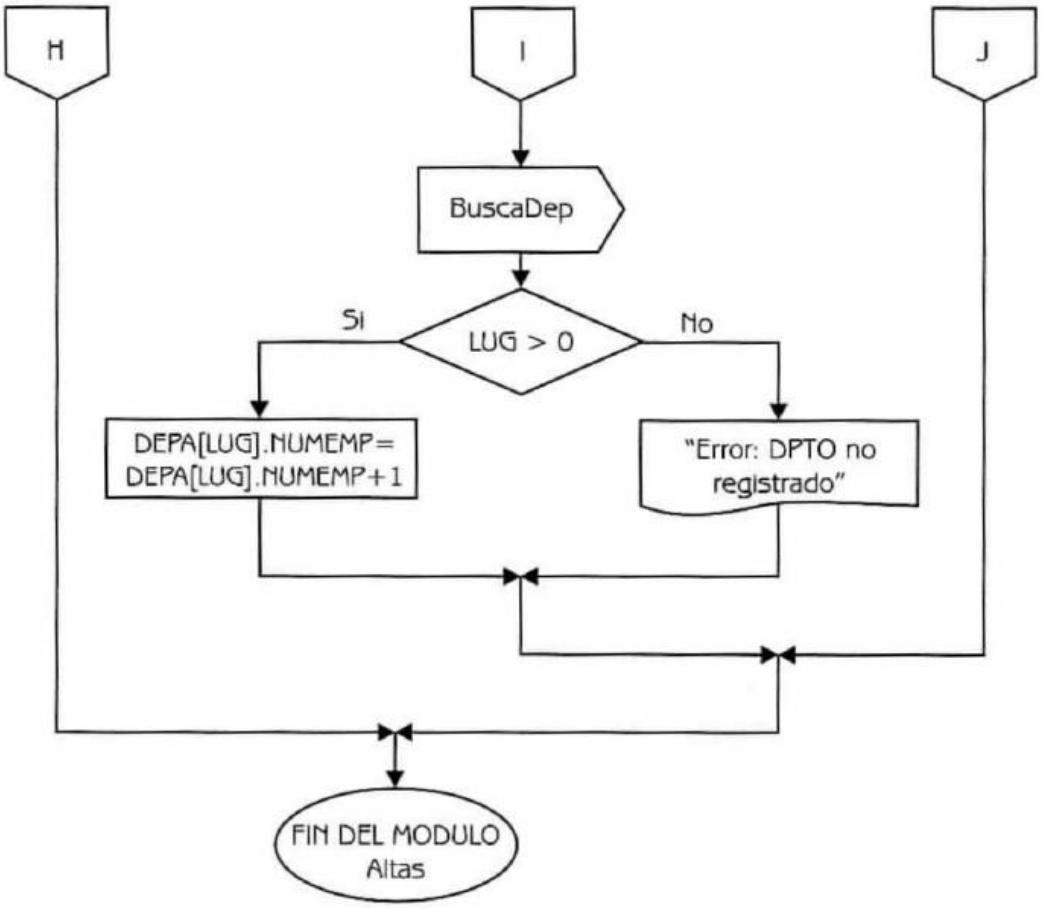


Diagrama de Flujo 5.4 (*continuación...*)

www.FreeLibros.com

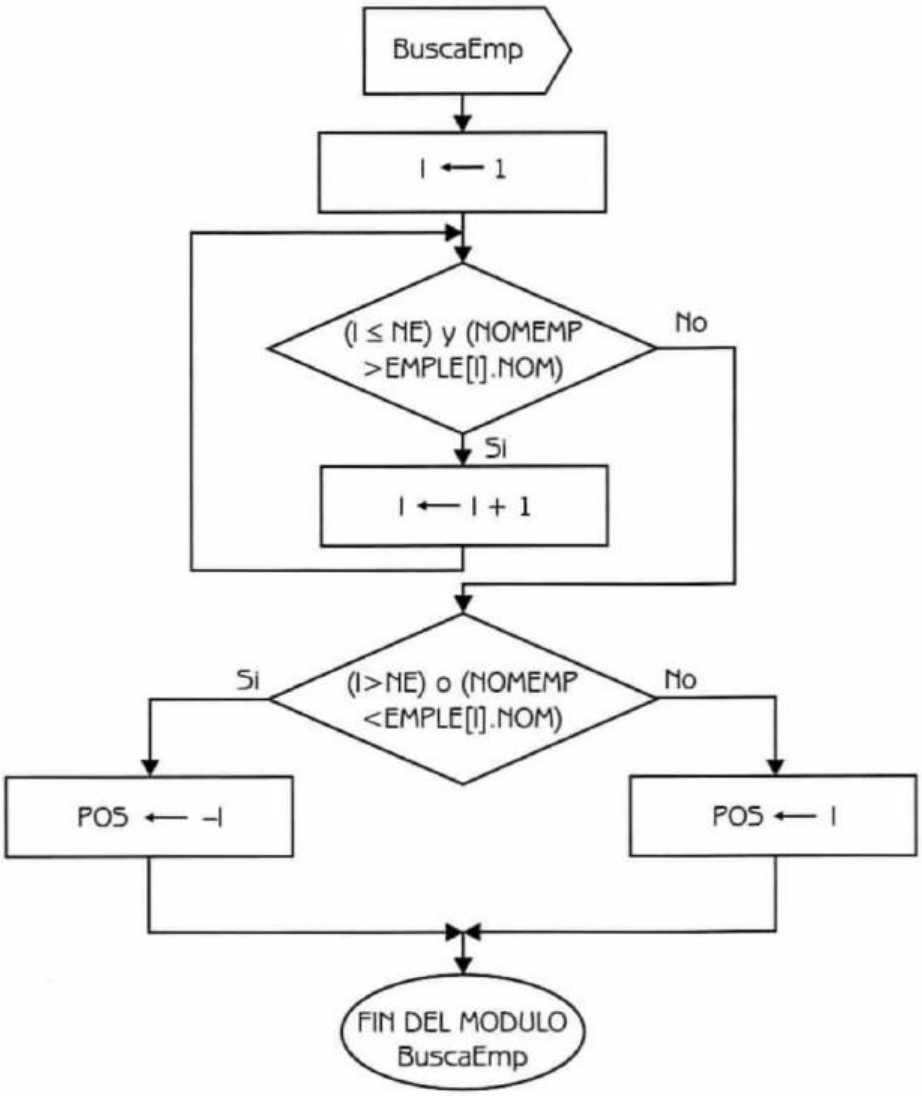


Diagrama de Flujo 5.4 (*continuación...*)

www.FreeLibros.com

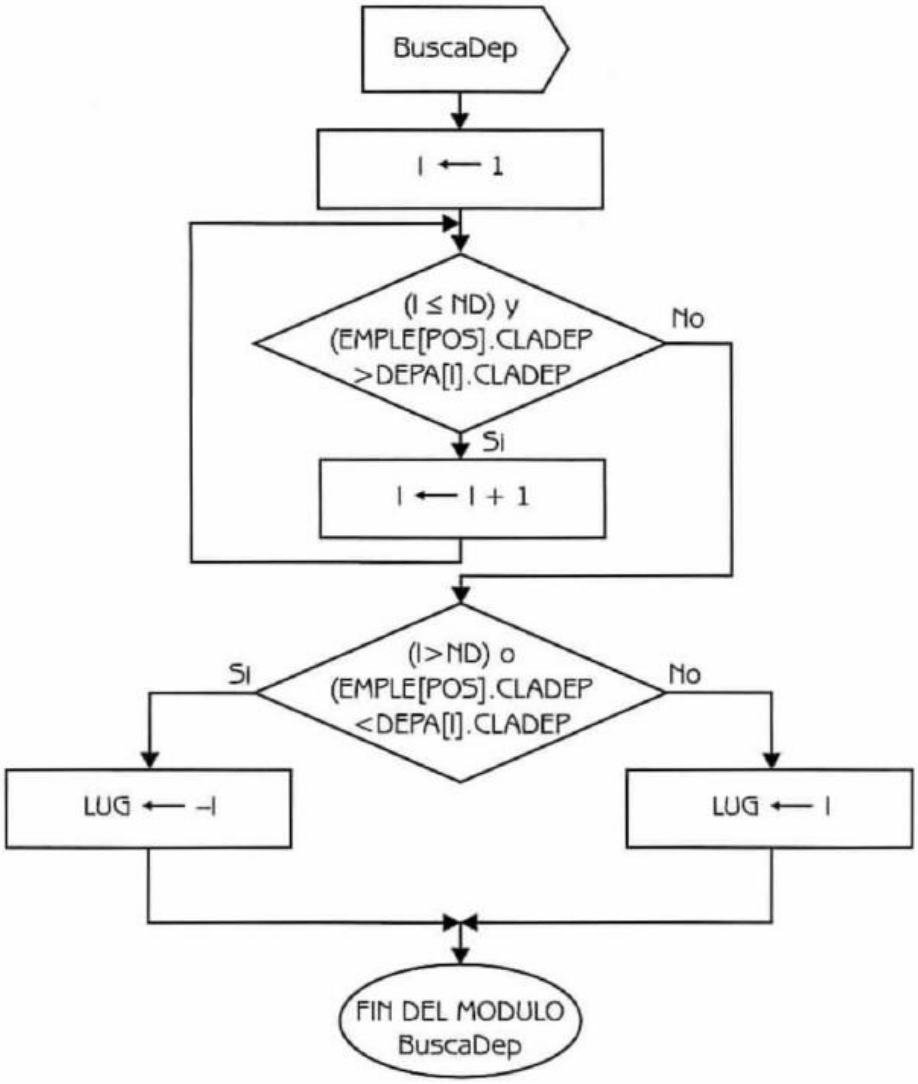


Diagrama de Flujo 5.4 (continuación)

Explicación de las variables

NE: Variable de tipo entero. Representa el número de empleados.

ND: Variable de tipo entero. Almacena el número de departamento.

I: Variable de tipo entero. Se usa como variable de control de varios ciclos y como índice de los arreglos.

EMPLE: Arreglo unidimensional de registros.

OPCION: Variable de tipo entero. Se usa para leer la opción seleccionada por el usuario.

DEPA: Arreglo unidimensional de registros.

www.FreeLibros.com

NOMEMP: Variable de tipo cadena de caracteres. Almacena el nombre del empleado del cual se solicita información (inciso a), o se dará de alta (inciso c).

POS: Variable de tipo entero. Almacena la posición en la que están o deberían estar registrados los datos de cierto empleado.

LUG: Variable de tipo entero. Almacena la posición en la que están o deberían estar registrados los datos de cierto departamento.

DEPARTA: Variable de tipo cadena de caracteres. Se usa para leer el nombre del departamento, del cual interesa conocer nombre y sueldo del empleado que más gana (inciso b).

MAX: Variable de tipo real. Se utiliza en la búsqueda del empleado con mayor sueldo en un departamento dado (inciso b).

K: Variable de tipo entero. Se usa como variable de control en el ciclo para realizar la búsqueda del sueldo más alto.

L: Variable de tipo entero. Auxilia en la búsqueda del sueldo más alto. Dado que se sabe el total de empleados por departamento, el uso de esta variable puede ayudar a evitar búsquedas innecesarias.

EMAX: Variable de tipo entero. Se usa para guardar la posición en la que se encontró el sueldo más alto.

CLAVE Variable de tipo entero. Se utiliza para almacenar la clave del departamento, dado el nombre del mismo.

Programa 5.4

```
EMPLEADOS_DEPARTAMENTO

{El programa, dada información sobre los empleados y los departamentos a los que
pertenece, realiza las siguientes operaciones: a) Dado el nombre de un empleado, pro-
porciona su sueldo, el nombre del departamento en que trabaja y el nombre de su
jefe, b) Dado el nombre de un departamento, obtiene el sueldo y el nombre del em-
pleado que más gana en ese departamento, y c) Dado el nombre de un nuevo em-
pleado, lo da de alta en los arreglos correspondientes}

{NE, ND, I, OPCION, POS, LUG, K, L y EMAX son variables de tipo entero. MAX es
una variable de tipo real. NOMEMP y DEPARTA son variables de tipo cadena de ca-
racteres. EMPLE y DEPA son arreglos unidimensionales de registros}

1. Escribir "Ingrese número de empleados [1..1000]"
```

www.FreeLibros.com

```
2. Leer NE
3. Si (NE >= 1) y (NE <= 1000)
     entonces
          Hacer I ⟵ 1
     3.1 Repetir con I desde 1 hasta NE
          Escribir "Ingrese nombre, clave dpto., antigüedad y sueldo del empleado", I
          Leer EMPLE[I].NOM, EMPLE[I].CLADEP, EMPLE[I].ANTI y
               EMPLE[I].SUE
          Hacer I ⟵ I + 1
     3.2 {Fin del ciclo del paso 3.1}
          Escribir "Ingrese número de departamentos [1..20]"
          Leer ND
     3.3 Si (ND >= 1) y (ND <= 20)
          entonces
             Hacer I ⟵ 1
       3.3.1 Repetir con I desde 1 hasta ND
             Escribir "Ingrese clave, nombre, total de empleados y nombre del
                       jefe del departamento", I
             Leer DEPA[I].CLADEP, DEPA[I].NOMDEP,
                  DEPA[I].NUMEMP y DEPA[I].NOMJEF
             Hacer I ⟵ I + 1
       3.3.2 {Fin del ciclo del paso 3.3.1}
             Escribir "Información sobre un empleado (1)",
                      "Empleado que más gana por dpto. (2)",
                      "Alta de empleados (3)",
                      "Terminar proceso (4)",
             Leer OPCION
       3.3.3 Mientras (OPCION >= 1) y (OPCION <= 3) repetir
         3.3.3.1 Si OPCION igual
                 1: Escribir "Nombre del empleado:?"
                    Leer NOMEMP
                    Hacer I ⟵ 1
           3.3.3.1.1 Mientras (I <= NE) y (NOMEMP > EMPLE[I].NOM) repetir
                          Hacer I ⟵ I + 1
           3.3.3.1.2 {Fin del ciclo del paso 3.3.3.1.1}
           3.3.3.1.3 Si (I > NE) o (NOMEMP < EMPLE[I].NOM)
                          entonces
                               Hacer POS ⟵ -I
                          sino
                               Hacer POS ⟵ I
           3.3.3.1.4 {Fin del condicional del paso 3.3.3.1.3}
           3.3.3.1.5 Si POS > 0
```

www.FreeLibros.com

```
            entonces
                Escribir NOMEMP, "gana", EMPLE[POS].SUE
                Hacer I ⟵ 1
    3.3.3.1.5.1 Mientras (I <= ND) y (EMPLE[POS].CLADEP
                        > DEPA[I].CLADEP) repetir
                    Hacer I ⟵ I + 1
    3.3.3.1.5.2 {Fin del ciclo del paso 3.3.3.1.5.1}
    3.3.3.1.5.3 Si (I > ND) o (EMPLE[POS].CLADEP
                    < DEPA[I].CLADEP)
                    entonces
                      Hacer LUG ⟵ -I
                    sino
                      Hacer LUG ⟵ I
    3.3.3.1.5.4 {Fin del condicional del paso 3.3.3.1.5.3}
    3.3.3.1.5.5 Si LUG < 0
                    entonces
                      Escribir "Error: CLADEP no registrada"
                    sino
                      Escribir NOMBRE, "trabaja en el dpto.",
                               DEPA[LUG].NOMDEP, "y su jefe
                               es", DEPA[LUG].NOMJEF
    3.3.3.1.5.6 {Fin del condicional del paso 3.3.3.1.5.5}
              sino
                Escribir "Error: Nombre de empleado no registrado"
  3.3.3.1.6 {Fin del condicional del paso 3.3.3.1.5}
        2: Escribir "Nombre del departamento:?"
           Leer DEPARTA
           Hacer I ⟵ 1
  3.3.3.1.7 Mientras (I <= ND) y (DEPARTA <> DEPA[I].NOMDEP)
           repetir
             Hacer I ⟵ I + 1
  3.3.3.1.8 {Fin del ciclo del paso 3.3.3.1.7}
  3.3.3.1.9 Si I > ND
           entonces
            Escribir "Error: Ese nombre de dpto. no está registrado"
           sino
            Hacer MAX ⟵ -1, K ⟵ 1 y CLAVE ⟵ DEPA[I].CLADEP
    3.3.3.1.9.1 Repetir con K desde 1 hasta NE
      3.3.3.1.9.1.1 Si (EMPLE[K].CLADEP=CLAVE)
                    entonces
        3.3.3.1.9.1.1.1 Si MAX < EMPLE[K].SUE entonces
                          Hacer MAX ⟵ EMPLE[K].SUE,
                                EMAX ⟵ K
```

www.FreeLibros.com

```
            3.3.3.1.9.1.1.2 {Fin del condicional del paso 3.3.3.1.9.1.1.1}
          3.3.3.1.9.1.2 {Fin del condicional del paso 3.3.3.1.9.1.1}
                        Hacer K ⟵ K + 1
        3.3.3.1.9.2 {Fin del ciclo del paso 3.3.3.1.9.1}
                    Escribir "Del dpto.", DEPARTA, "el empleado que más gana
                              es", EMPLE[EMAX].NOM, "y su sueldo es", MAX
      3.3.3.1.10 {Fin del condicional del paso 3.3.3.1.9}
              3: Escribir "Nombre del nuevo empleado:"
                 Leer NOMEMP
                 Hacer I ⟵ 1
      3.3.3.1.11 Mientras (I <= NE) y (NOMEMP > EMPLE[I].NOM) repetir
                        Hacer I ⟵ I + 1
      3.3.3.1.12 {Fin del ciclo del paso 3.3.3.1.11}
      3.3.3.1.13 Si (I > NE) o (NOMEMP < EMPLE[I].NOM)
                        entonces
                            Hacer POS ⟵ -I
                        sino
                            Hacer POS ⟵ I
      3.3.3.1.14 {Fin del condicional del paso 3.3.3.1.13}
      3.3.3.1.15 Si POS > 0
                        entonces
                            Escribir "Error: Ese nombre ya está registrado"
                        sino
        3.3.3.1.15.1 Si NE < 1000
                            entonces
                              Hacer POS ⟵ POS * (- 1), NE ⟵ NE + 1 e
                                        I ⟵ NE
          3.3.3.1.15.1.1 Repetir con I desde NE hasta (POS + 1)
                              Hacer EMPLE[I] ⟵ EMPLE[I - 1] e I ⟵ I - 1
          3.3.3.1.15.1.2 {Fin del ciclo del paso 3.3.3.1.15.1.1}
                        Escribir "Ingrese clave depto., antigüedad y sueldo
                                  del nuevo empleado"
                              Leer EMPLE[POS].CLADEP,
                                   EMPLE[POS].ANTI y EMPLE[POS].SUE
                              Hacer EMPLE[POS].NOM ⟵ NOMEMP
                              Hacer I ⟵ 1
          3.3.3.1.15.1.3 Mientras (I <= ND) y (EMPLE[POS].CLADEP
                                  > DEPA[I].CLADEP) repetir
                              Hacer I ⟵ I + 1
          3.3.3.1.15.1.4 {Fin del ciclo del paso 3.3.3.1.15.1.3}
          3.3.3.1.15.1.5 Si (I > ND) o (EMPLE[POS].CLADEP
                              < DEPA[I].CLADEP)
                              entonces
```

www.FreeLibros.com

```
                                        Hacer LUG ⟵ -I
                                    sino
                                        Hacer LUG ⟵ I
            3.3.3.1.15.1.6 {Fin del condicional del paso 3.3.3.1.15.1.5}
            3.3.3.1.15.1.7 Si LUG > 0
                                    entonces
                                        Hacer DEPA[LUG].NUMEMP
                                        ⟵ DEPA[LUG].NUMEMP + 1
                                    sino
                                        Escribir "Error: Dpto. no registrado"
            3.3.3.1.15.1.8 {Fin del condicional del paso 3.3.3.1.15.1.7}
                                sino
                                    Escribir "Error: No hay espacios para altas"
          3.3.3.1.15.2 {Fin del condicional del paso 3.3.3.1.15.1}
        3.3.3.1.16 {Fin del condicional del paso 3.3.3.1.15}
      3.3.3.2 {Fin del condicional del paso 3.3.3.1}
              Escribir "Ingrese opción de operación"
              Leer OPCION
    3.3.4 {Fin del ciclo del paso 3.3.3}
          Escribir "Datos actualizados"
          Hacer I ⟵ 1
    3.3.5 Repetir con I desde 1 hasta NE
              Escribir "Nombre:", EMPLE[I].NOM,
                       "Clave dpto.:", EMPLE[I].CLADEP,
                       "Antigüedad:", EMPLE[I].ANTI,
                       "Sueldo:", EMPLE[I].SUE
              Hacer I ⟵ I + 1
    3.3.6 {Fin del ciclo del paso 3.3.5}
          Hacer I ⟵ 1
    3.3.7 Repetir con I desde 1 hasta ND
              Escribir "Clave:", DEPA[I].CLADEP,
                       "Nombre.:", DEPA[I].NOMDEP,
                       "Total de empleados:", DEPA[I].NUMEMP
                       "Nombre del jefe:", DEPA[I].NOMJEF
              Hacer I ⟵ I + 1
    3.3.8 {Fin del ciclo del paso 3.3.7}
      sino
          Escribir "Error"
  3.4 {Fin del condicional del paso 3.3}
    sino
      Escribir "Error"
4. {Fin del condicional del paso 3}
```

www.FreeLibros.com

6

Problemas suplementarios

PS 1.1

Escriba un diagrama de flujo tal que dado como datos dos números reales, calcule la suma, resta y multiplicación de dichos números.

Datos: N1, N2 (variables de tipo real que representan los números reales).

PS 1.2

Construya un diagrama de flujo tal que dado el radio de un círculo, calcule e imprima el área y la circunferencia.

Dato: RADIO (variable de tipo real que representa el radio del círculo).

Consideraciones:

- El área de un círculo la calculamos como:

$$\text{Area} = \pi * \text{radio}^2$$

Fórmula 6.1

- La circunferencia la calculamos como:

$$\text{Circunferencia} = 2 * \pi * \text{radio}$$

Fórmula 6.2

www.FreeLibros.com

PS 1.3

En una Casa de Cambio necesitan construir un programa tal que dado como dato una cantidad expresada en dólares, convierta esa cantidad a pesos. Construya el diagrama de flujo correspondiente.

Datos: CAN (variable real que representa una cantidad en dólares).

Consideraciones:

- Observe que el tipo de cambio actual es el siguiente:

1 dólar ⟵ 11.96 pesos

PS 1.4

Una persona compró una estancia en un país sudamericano. La extensión de la estancia está especificada en acres. Construya un diagrama de flujo tal que dado como dato la extensión del campo en "acres", calcule e imprima la extensión del mismo en hectáreas.

Datos: ECA (variable de tipo real que especifica la extensión del campo en acres).

Consideraciones:

Recuerde lo siguiente:

- 1 acre es igual a 4047 m^2.
- 1 hectárea tiene 10000 m^2.

PS 1.5

Una persona invierte en un banco una determinada cantidad de dinero y a una cierta tasa de interés mensual. Construya un diagrama de flujo que permita obtener el monto del dinero que obtendrá al finalizar el mes.

Datos: MD, TASA

Donde:

MD es una variable de tipo real que representa el monto del dinero que colocará la persona.

TASA es una variable de tipo real que señala la tasa de interés mensual.

www.FreeLibros.com

PS 1.6

Dada la estabilidad económica que existe en un determinado país de América Latina, las agencias automotrices comienzan a ofrecer distintos planes de financiamiento para la comercialización de sus vehículos. La empresa XGW ofrece el siguiente plan de financiación: dado el monto total del vehículo, el cliente debe pagar el 35% de enganche y el resto en 18 mensualidades iguales sin intereses. Construya el diagrama de flujo que permita obtener cuál es el importe del enganche y las mensualidades que debe pagar el cliente.

Dato: MON (variable de tipo real que representa el precio del vehículo).

PS 1.7

La misma empresa comercializadora de vehículos XGW ofrece planes de financiación hasta 36 meses con un enganche del 35%, pero aplicando al saldo restante una tasa de interés global del 12%. Construya el diagrama de flujo que permita obtener tanto el importe del enganche como el de las mensualidades que debe pagar el cliente.

Dato: MON (variable de tipo real que representa el precio del vehículo).

PS 1.8

Construya un diagrama de flujo tal que dado el radio, la generatriz y la altura de un cono, calcule e imprima el área de la base, el área lateral, el área total y su volumen.

Datos: RADIO, ALTU, GENE

Donde:

RADIO	es una variable de tipo real que representa el radio del cono.
ALTU	es una variable de tipo real que representa la altura del cono.
GENE	es una variable de tipo real que representa la generatriz.

Consideraciones:

- Un cono tiene la siguiente forma:

www.FreeLibros.com

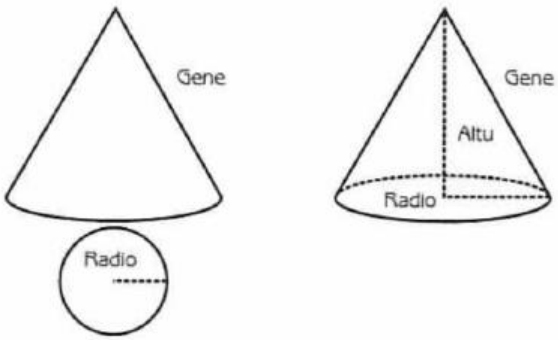


Figura 6.1 Cono

- El área de la base se calcula aplicando la siguiente fórmula:

$$AB = \pi * RADIO^2$$

Fórmula 6.3

- El área lateral se calcula:

$$AL = \pi * RADIO * GENE$$

Fórmula 6.4

- El área total se calcula como:

$$AT = AB + AL$$

Fórmula 6.5

- El volumen se calcula de esta forma:

$$VOL = \frac{1}{3} * AB * ALTU$$

Fórmula 6.6

PS 1.9

Construya un diagrama de flujo tal que dado el radio de una esfera, calcule e imprima el área y su volumen.

Dato: RADIO (variable de tipo real que representa el radio de la esfera).

Consideraciones:

- Una esfera tiene la siguiente forma:

www.FreeLibros.com

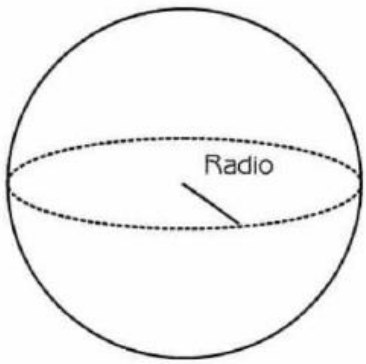


Figura 6.2 Esfera

- El área de una esfera la calculamos de esta forma:

$$\text{AREA} = 4 * \pi * \text{RADIO}^2$$

Fórmula 6.7

- El volumen como:

$$\text{VOL} = \frac{4}{3} * \pi * \text{RADIO}^3$$

Fórmula 6.8

PS 1.10

Construya un diagrama de flujo tal que dado como dato el lado de un *hexaedro o cubo,* calcule el área de la base, el área lateral, el área total y el volumen.

Datos: L (variable real que representa el lado de un hexaedro o cubo).

Consideraciones:

- Un hexaedro o cubo tiene la siguiente forma:

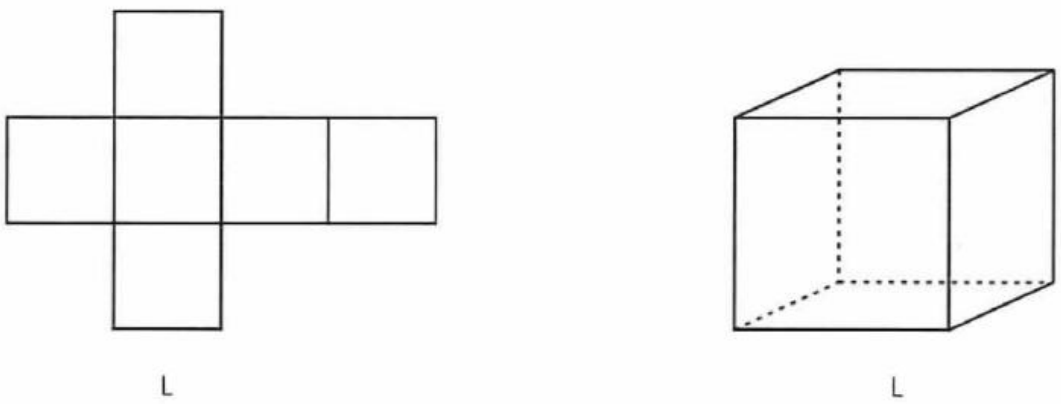


Figura 6.3 Hexaedro o Cubo

www.FreeLibros.com

- Para calcular el área de la base aplicamos la siguiente fórmula:

$$AB = L^2$$

Fórmula 6.9

- Para calcular el área lateral hacemos:

$$AL = 4 * L^2$$

Fórmula 6.10

- Para calcular el área total hacemos:

$$AT = 6 * L^2$$

Fórmula 6.11

- Para calcular el volumen hacemos:

$$V = L^3$$

Fórmula 6.12

PS 1.11

Construya un diagrama de flujo tal que dadas las coordenadas de los puntos P1, P2 y P3 que corresponden a los vértices de un triángulo, calcule su perímetro.

Datos: X1, Y1, X2, Y2, X3, Y3

Donde:

X1 y Y1 son variables de tipo real que representan las coordenadas en el eje de las X y las Y, del punto P1.

X2 y Y2 son variables de tipo real que expresan las coordenadas en el eje de las X y las Y, del punto P2.

X3 y Y3 son variables de tipo real que representan las coordenadas en el eje de las X y las Y, del punto P3.

Consideraciones:

Para calcular la distancia entre dos puntos P1 y P2 hacemos:

www.FreeLibros.com

$$D = \sqrt{(X1 - X2)^2 + (Y1 - Y2)^2}$$

Fórmula 6.13

PS 1.12

Construya un diagrama de flujo tal que dadas las coordenadas de los puntos P1, P2 y P3 que corresponden a los vértices de un triángulo, calcule su superficie.

Datos: X1, Y1, X2, Y2, X3, Y3

Donde:

X1 y Y1 son variables de tipo real que representan las coordenadas en el eje de las X y las Y, del punto P1.

X2 y Y2 son variables de tipo real que expresan las coordenadas en el eje de las X y las Y, del punto P2.

X3 y Y3 son variables de tipo real que representan las coordenadas en el eje de las X y las Y, del punto P3.

Consideraciones:

- Para calcular el área de un triángulo dadas las coordenadas de los vértices que la componen, debemos aplicar la siguiente fórmula:

$$AREA = \frac{1}{2} * \left| X1 * (Y2 - Y3) + X2 * (Y3 - Y1) + X3 * (Y1 - Y2) \right|$$

Fórmula 6.14

O bien, esta otra:

$$AREA = \frac{1}{2} * \left| (X2 - X1) * (Y3 - Y1) - (X3 - X1) * (Y2 - Y1) \right|$$

Fórmula 6.15

www.FreeLibros.com

PS 1.13

Construya un diagrama de flujo tal que dado el perímetro de la base, el apotema y la altura de un prisma pentagonal; calcule el área de la base, el área lateral, el área total y el volumen.

Datos: PER, APO, ALT

Donde:

PER	es una variable de tipo real que representa el perímetro de la base.
APO	es una variable de tipo real que representa el apotema.
ALT	es una variable de tipo real que expresa la altura del prisma pentagonal.

Consideraciones:

Un prisma pentagonal tiene la siguiente forma:

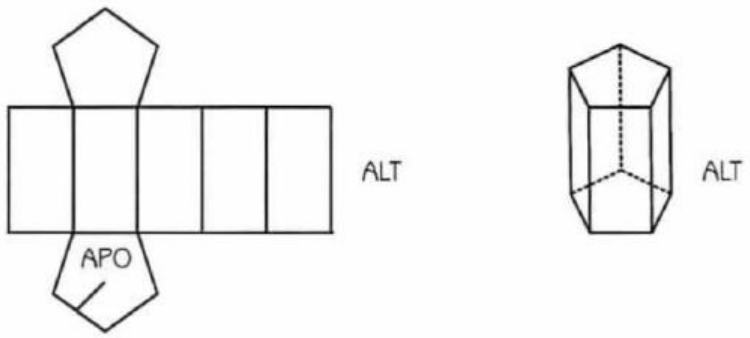


Figura 6.4 Prisma Pentagonal

- Para calcular el área de la base, hacemos:

$$AB = \frac{PER * APO}{2}$$

Fórmula 6.16

- Para calcular el área lateral, aplicamos la siguiente fórmula:

$$AL = PER * ALT$$

Fórmula 6.17

- Para calcular el área total hacemos:

www.FreeLibros.com

$$AT = 2 * AB + AL$$

Fórmula 6.18

- Para calcular el volumen hacemos:

$$VOL = AB * ALT$$

Fórmula 6.19

www.FreeLibros.com

PS 2.1

La tangente de un ángulo se define como el cociente entre el seno y el coseno de dicho ángulo.

$$\tan(\alpha) = \frac{\text{sen}(\alpha)}{\cos(\alpha)}$$

Fórmula 6.20

Construya un diagrama de flujo que le permita calcular la tangente de un ángulo, considerando que se conoce el valor del seno y del coseno del mismo. Recuerde que el coseno debe ser diferente de 0.

Datos: SENO, COSENO (variables de tipo real).

PS 2.2

La cotangente de un ángulo se define como el cociente entre el coseno y el seno de dicho ángulo.

$$\text{cot g}(\alpha) = \frac{\cos(\alpha)}{\text{sen}(\alpha)}$$

Fórmula 6.21

Construya un diagrama de flujo que le permita calcular la cotangente de un ángulo, considerando que se conoce el valor del seno y del coseno del mismo. Recuerde que el seno debe ser diferente de 0.

Datos: SENO, COSENO (variables de tipo real).

PS 2.3

En un negocio de productos electrodomésticos aplican un descuento del 8% a todos aquellos clientes cuya compra es superior a $ 2,500. Dado como dato el monto de compra del cliente, calcule lo que el mismo debe pagar. Haga el diagrama de flujo correspondiente.

Dato: COMPRA (variable de tipo real que representa la compra efectuada por el cliente).

www.FreeLibros.com

PS 2.4

Dado como dato el sueldo de un trabajador, considere un aumento del 15% si su sueldo es inferior a $1,000 y de un 12% en caso contrario. Imprima el sueldo con el aumento incorporado. Haga el diagrama de flujo correspondiente

Dato: SUE (variable real que representa el sueldo del trabajador).

PS 2.5

Dados los datos A, B, C y D que representan números enteros, construya un diagrama de flujo que calcule el resultado de las siguientes expresiones (Si D es igual a 0 imprima el mensaje que considere apropiado):

$$\frac{(A-C)^2}{D} \quad \text{y} \quad \frac{(A-B)^3}{D}$$

Datos: A, B, C y D (variables de tipo entero).

PS 2.6

Haga un diagrama de flujo para calcular el precio del billete ida y vuelta en ferrocarril, conociendo la distancia del viaje de ida y el tiempo de estancia. Se sabe además que si el número de días de estancia es superior a 7 y la distancia total (ida y vuelta) a recorrer es superior a 800 km, el billete tiene una reducción del 30%. El precio por km es de $0.23.

Datos: DIST, TIEM.

Donde:

DIST es una variable de tipo entero que representa la distancia del viaje de ida.
TIEM es una variable de tipo entero que representa el tiempo de estancia.

PS 2.7

Construya un diagrama de flujo tal que dado como dato el sueldo de un trabajador, calcule su aumento según el siguiente criterio:

- SUELDO $<$ $10,000 $\Rightarrow$ AUMENTO 15%
- $10,000 $\leq$ SUELDO $\leq$ $15,000 $\Rightarrow$ AUMENTO 11%
- SUELDO $>$ $15,000 $\Rightarrow$ AUMENTO 8%

www.FreeLibros.com

Imprima el nuevo sueldo del trabajador.

Dato: SUELDO (variable de tipo real que representa el sueldo del trabajador).

PS 2.8

Construya un diagrama de flujo que pueda determinar, dados dos números enteros, si un número es divisor de otro.

Datos: NUM1, NUM2 (variables de tipo entero).

PS 2.9

Construya un diagrama de flujo tal que dado como datos A y N, compruebe la igualdad de la siguiente expresión:

$$A^{-N} = \frac{1}{A^N} \qquad (A \neq 0)$$

Datos: A, N (variables de tipo entero).

PS 2.10

Construya un diagrama de flujo tal que dado como datos A, B y N, compruebe la igualdad de la siguiente expresión:

$$\left(\frac{A}{B}\right)^N = \frac{A^N}{B^N}$$

Datos: A, B, N (variables de tipo entero).

PS 2.11

Construya un diagrama de flujo tal que dado como dato Y, calcule el resultado de la siguiente función:

$$X = \begin{cases} 3 = Y + 36 & \text{si } 0 < Y \leq 11 \\ Y^2 - 10 & \text{si } 11 < Y \leq 33 \\ Y^3 + Y^2 - 1 & \text{si } 33 < Y \leq 64 \\ 0 & \text{Para cualquier otro valor de Y} \end{cases}$$

Imprima X y Y.

www.FreeLibros.com

Dato: Y (variable de tipo real).

PS 2.12

Construya un diagrama de flujo tal que dado como dato una temperatura en grados Fahrenheit, determine el deporte que es apropiado practicar a esa temperatura, teniendo en cuenta la siguiente tabla:

Tabla 6.1

DEPORTE	TEMPERATURA
Natación	> 85
Tenis	$70 < \text{TEMP} \leq 85$
Golf	$32 < \text{TEMP} \leq 70$
Esquí	$10 < \text{TEMP} \leq 32$
Marcha	≤ 10

Dato: TEMP (variable de tipo real que representa la temperatura).

PS 2.13

Dados tres números reales A, B y C, identifique cúal es el mayor. Considere a fin de simplificar el problema que los números son diferentes. Haga el diagrama de flujo correspondiente.

Datos: A, B, C (variables de tipo real).

PS 2.14

Haga un diagrama de flujo que permita calcular el valor de f(x), según la expresión:

$$f(x) = \begin{cases} x^2 & \text{si } (x \bmod 4) = 0 \\ x/6 & \text{si } (x \bmod 4) = 1 \\ \sqrt{x} & \text{si } (x \bmod 4) = 2 \\ x^3 + 5 & \text{si } (x \bmod 4) = 3 \end{cases}$$

Dato: X (variable de tipo entero).

PS 2.15

Construya un diagrama de flujo que permita realizar operaciones aritméticas elementales, según la clave ingresada.

www.FreeLibros.com

Tabla 6.2	
CLAVE	**OPERACION**
+	SUMA
-	RESTA
*	MULTIPLICACION
/	DIVISION

Imprima la clave ingresada y el resultado de la operación.

Datos: OPER1, OPER2, CLAVE

Donde:

OPER1 es una variable de tipo real que representa el primer operando.
OPER2 es una variable de tipo real que expresa el segundo operando.
CLAVE es una variable de tipo caracter que representa el tipo de operación aritmética que se va a realizar.

PS 2.16

Construya un diagrama de flujo tal que dado como datos el modelo de un vehículo y su precio, determine el valor final que debe pagar el comprador. El concesionario está haciendo descuentos teniendo en cuenta el modelo, con base en la siguiente tabla.

Tabla 6.3	
MODELO	**DESCUENTO**
Blazer-Trail	8%
Cavalier	5%
Chevy	6%
Opel-Astra	9%

Datos: MODELO, PRECIO

Donde:

MODELO es una variable de tipo cadena de caracteres que representa el modelo del vehículo.
PRECIO es una variable de tipo real que representa el precio del vehículo.

PS 2.17

Escriba un diagrama de flujo que permita convertir de pulgadas a milímetros, de yardas a metros y de millas a kilómetros.

www.FreeLibros.com

Datos: OPCION, MED

Donde:

OPCION es una variable de tipo entero que representa el tipo de conversión que se desea realizar. Se ingresa:

1. Para convertir de pulgadas a milímetros.
2. Para convertir de yardas a metros.
3. Para convertir de millas a kilómetros.

MED es una variable de tipo real que representa la medida que se ingresará para conversión.

Consideraciones:

- pulgada equivale a 25.40 milímetros.
- 1 yarda equivale a 0.9144 metros.
- 1 milla equivale a 1.6093 kilómetros.

PS 2.18

Construya un diagrama de flujo que le permita calcular e imprimir el costo final de una llamada telefónica. Para calcular el costo final se sigue lo indicado en la siguiente tabla:

Tabla 6.4

CLAVE	ZONA	PRECIO/MINUTO (3 primeros)	PRECIO/MINUTO (del 4° en adelante)
12	América del Norte	2	1.5
15	América Central	2.2	1.8
18	América del Sur	4.5	3.5
19	Europa	3.5	2.7
23	Asia	6	4.6
25	Africa	6	4.6
29	Oceanía	5	3.9

Datos: CLAVE, NUMIN

Donde:

CLAVE es una variable entera que representa la clave de la zona geográfica a la que se llamó.

www.FreeLibros.com

NUMIN es una variable entera que significa la duración (en minutos) de la llamada.

PS 2.19

Dado tres datos enteros positivos que representan las longitudes de los lados de un probable triángulo, construya un diagrama de flujo que determine efectivamente si los datos corresponden a un triángulo. En caso de que sí correspondan, escriba si el triángulo es equilátero, isósceles o escaleno. Calcule además su área.

Considere que es triángulo, si se cumple que la suma de los dos lados menores es mayor que la del lado mayor. Tome en cuenta además que el área de un triángulo la calculamos como:

$$\text{AREA} = (S\,(S-A)\,(S-B)\,(S-C))^{1/2}$$

Fórmula 6.22

Donde:

S representa la mitad de la suma de los lados A, B, C.

Datos: A, B, C (variables de tipo entero que representan los lados del posible triángulo).

PS 2.20

Escriba un diagrama de flujo que permita realizar la conversión de pesos y medidas. Considere las siguientes equivalencias:

Medidas de longitud

1 pulgada equivale a 25.40 milímetros
1 yarda equivale a 0.9144 metro
1 milla equivale a 1.6093 kilómetro
1 pulgada^2 equivale a 6.452 centímetros^2
1 pie^2 equivale a 0.09290 metro^2
1 yarda^2 equivale a 0.8361 metro^2
1 acre equivale a 0.4047 hectárea
1 milla^2 equivale a 2.59 kilómetros^2

Medidas de volumen

1 pie^3 equivale a 0.02832 metro^3
1 yarda^3 equivale a 0.7646 metro^3
1 pinta equivale a 0.56826 litro
1 galón equivale a 4.54609 litros

www.FreeLibros.com

Medidas de peso

1 onza equivale a 28.35 gramos
1 libra equivale a 0.45359 kilogramo
1 ton inglesa equivale a 1.0160 tonelada

Dato: OPCION, SUBOPC, MED

Donde:

OPCION es una variable entera que representa el tipo de medida que se desea convertir. Se ingresa:

1: Para medidas de longitud
2: Para medidas de volumen
3: Para medidas de peso

SUBOPC es una variable de tipo cadena de caracteres que representa dentro de un tipo de medida, el tipo de conversión que se desea realizar. Se ingresa:

L1: Para convertir de pulgadas a milímetros
L2: Para convertir de yardas a metros
L3: Para convertir de millas a kilómetros
L4: Para convertir de pulgadas^2 a centímetros^2
L5: Para convertir de pies^2 a metros^2
L6: Para convertir de yardas^2 a metros^2
L7: Para convertir de acres a hectáreas
L8: Para convertir de millas^2 a kilómetros^2

V1: Para convertir de pies^3 a metros^3
V2: Para convertir de yardas^3 a metros^3
V3: Para convertir de pintas a litros
V4: Para convertir de galón a litros

P1: Para convertir de onzas a gramos
P2: Para convertir de libras a kilogramos
P3: Para convertir de ton inglesa a toneladas

MED es una variable de tipo real que expresa la medida que se ingresa para conversión.

PS 2.21

Los clientes de un hospital tienen una credencial en la que además de otra información registra una categoría que depende de los ingresos económicos del nú-

www.FreeLibros.com

cleo familiar del paciente (cliente). Teniendo en cuenta la categoría el hospital les hace un descuento cuando tienen que pagar su cuenta, con base en la siguiente tabla:

Tabla 6.5	
CATEGORIA	DESCUENTO
1	35 %
2	22 %
3	15 %
4	5 %

Construya el diagrama de flujo y programa correspondiente que resuelva el problema. Observe que las categorías mayor a 4 no tienen descuento.

Datos: CATE, MONTO

Donde:

CATE es una variable de tipo entero que representa la categoría del cliente.

MONTO es una variable de tipo real que señala lo que debe pagar el cliente (obviamente sin considerar el descuento correspondiente).

www.FreeLibros.com

PS 3.1

Construya un diagrama de flujo que calcule e imprima la suma de los N primeros números naturales.

Dato: N (variable de tipo entero).

PS 3.2

Se tienen las calificaciones de un grupo de alumnos que presentaron un examen. El profesor desea obtener el promedio de estas calificaciones. Escriba un diagrama de flujo para resolver lo planteado anteriormente.

Datos: $CAL_1, CAL_2, CAL_3, \ldots, -1$

Donde:

CALi es una variable de tipo real que representa la calificación del alumno i. El fin de datos está dado por –1.

PS 3.3

Escriba un diagrama de flujo tal que dado como datos N números enteros, determine cuántos de ellos son pares y cuántos impares.

Datos: $N, NUM_1, NUM_2, \ldots, NUM_N$

Donde:

N es una variable de tipo entero que representa el número de enteros que se ingresan.

NUMi es una variable de tipo entero que representa el número i que se ingresa ($1 \leq i \leq N$).

PS 3.4

Haga un diagrama de flujo para obtener la tabla de multiplicar de un número entero K, comenzando desde 1.

Dato: K (variable de tipo entero que representa el número entero del cual queremos obtener la tabla de multiplicar).

PS 3.5

Escriba un diagrama de flujo que lea un número entero N y calcule el resultado de la siguiente serie:

www.FreeLibros.com

$$1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \cdots + \frac{1}{N}$$

Datos: N (variable de tipo entero que representa el número de términos de la serie).

PS 3.6

Dado el sueldo de N trabajadores, considere un aumento del 15% a cada uno de ellos si su sueldo es inferior a $800. Imprima el sueldo con el aumento incorporado (si corresponde). Haga el diagrama de flujo correspondiente.

Datos: N, SUE_1, SUE_2, . . ., SUE_N

Donde:

N es una variable de tipo entero que representa el número de empleados de la empresa.
SUEi es una variable de tipo real que representa el sueldo del trabajador i ($1 \leq i \leq N$).

PS 3.7

Calcule el aumento de sueldos para N empleados de una empresa, bajo el siguiente criterio:

- Si el sueldo es menor a $10,000 : Aumento 10%
- Si el sueldo está comprendido entre $10,000 y $25,000 : Aumento 7%
- Si el sueldo es mayor a $25,000 : Aumento 8%

Imprima lo siguiente:

a) El nuevo sueldo del trabajador
b) El monto total de la nómina considerando el aumento.

Datos: N, SUE_1, SUE_2, . . ., SUE_N

Donde:

N es una variable de tipo entero que representa el número de empleados de la empresa.
SUE_i. es una variable de tipo real que representa el sueldo del trabajador i ($1 \leq i \leq N$).

www.FreeLibros.com

PS 3.8

Construya un diagrama de flujo que lea 100 números naturales y cuente cuántos de ellos son positivos, negativos o nulos.

Datos: $NUM_1, NUM_2, \ldots, NUM_N$ (variable de tipo entero que representa al número natural i, $1 \leq i \leq N$).

PS 3.9

Haga un diagrama de flujo que calcule la suma de los números pares comprendidos entre 10 y 50.

PS 3.10

Haga un diagrama de flujo que calcule e imprima la productoria de los N primeros números naturales.

$$\prod_{i=1}^{N} i$$

Dato: N (variable de tipo entero).

PS 3.11

Escriba un diagrama de flujo tal que dado el peso, la altura y el sexo de N personas que pertenecen a un estado de la república, obtenga tanto el promedio del peso como de la altura de esta población.

Datos: $N, PES_1, ALT_1, SEX_1, PES_2, ALT_2, SEX_2, \ldots, PES_N, ALT_N, SEX_N$

Donde:

- N es una variable de tipo entero que representa el número de personas que se considerarán en la muestra.
- PES_i es una variable de tipo real que representa el peso de la persona i $(1 \leq i \leq N)$.
- ALT_i es una variable de tipo real que representa la altura de la persona i $(1 \leq i \leq N)$.
- SEX_i es una variable de tipo entero que expresa el sexo de la persona i. Se ingresa 0 si es hombre y 1 si es mujer $(1 \leq i \leq N)$.

www.FreeLibros.com

PS 3.12

Resuelva el problema anterior, pero considere además el promedio de la altura y del peso en función del sexo. Es decir, nos interesa obtener el promedio de la altura y peso tanto de la población femenina como de la masculina.

Datos: $N, PES_1, ALT_1, SEX_1, PES_2, ALT_2, SEX_2, \ldots, PES_N, ALT_N, SEX_N$

PS 3.13

Se tienen las calificaciones de un grupo de alumnos que presentaron un examen de computación. Haga un diagrama de flujo que calcule e imprima cuántas calificaciones hay en cada uno de los siguientes rangos:

0 . . . 3.99
4 . . . 5.99
6 . . . 7.99
8 . . . 10

Datos: $CAL_1, CAL_2, \ldots, -1$

Donde:

CAL_i es una variable de tipo real que representa la calificación del alumno i. El fin de datos está dado por –1.

PS 3.14

Una persona invierte en un banco un cierto capital y quiere saber cuánto obtendrá al cabo de cierto tiempo, si el dinero se colocó a una determinada tasa de interés mensual. Haga el diagrama de flujo correspondiente.

Datos: MESES, CAPINI, $TASA_1, TASA_2, \ldots, TASA_{meses}$

Donde:

MESES es una variable de tipo entero que representa el número de meses al que se colocará la inversión.

CAPINI es una variable de tipo real que representa el capital inicial que se invertirá.

$TASA_i$ es una variable de tipo real que significa la tasa de interés del mes i ($1 \leq i \leq$ MESES).

www.FreeLibros.com

PS 3.15

Construya un diagrama de flujo que imprima todos los números de la secuencia FIBONACCI, mientras que el número no exceda de 50 000. La impresión debe ser de esta forma:

```
1 - 0
2 - 1
3 - 1
4 - 2
5 - 3
6 - 5
7 - 8
8 - 13
...
```

PS 3.16

Escriba un diagrama de flujo que reciba como entrada 24 números reales que representan las temperaturas del exterior en un período de 24 horas. Encuentre la temperatura media, asi como la más alta y más baja del día.

Datos: $TEMP_1, TEMP_2, \ldots TEM_{24}$

Donde:

$TEMP_i$ es una variable de tipo real que representa la temperatura de la hora i que se ingresa ($1 \leq i \leq 24$).

PS 3.17

Dado N valores de Y, haga un diagrama de flujo para calcular el resultado de la siguiente función:

$$X = \begin{cases} 3 * Y + 36 & \text{si } 0 < Y \leq 11 \\ Y^4 - 10 & \text{si } 11 < Y \leq 33 \\ Y^{15} + Y^{10} - 1 & \text{si } 33 < Y \leq 64 \\ 0 & \text{Para cualquier otro valor de Y} \end{cases}$$

Datos: $N, Y_1, Y_2, \ldots Y_N$

Donde:

N es una variable de tipo entero que representa el número de Y que se ingresarán.

www.FreeLibros.com

Yi es una variable de tipo real que representa el valor de la i–ésima Y, que se ingresa ($1 \leq i \leq N$).

Resultados: Imprima lo siguiente.

Y_1, X_1
Y_2, X_2
. .
. .
. .
Y_N, X_N

PS 3.18

Resuelva el problema anterior, pero ahora considere que no existe la operación de potencia.

Datos: N, Y_1, Y_2,. . . , Y_N

PS 3.19

Resuelva el problema anterior, pero ahora considere que no existen ni la operación de potencia ni la de multiplicación.

Datos: N, Y_1, Y_2,. . ., Y_N

PS 3.20

La compañía de Luz "Del Centro" desea imprimir los recibos de cobro de los cuentahabientes en el último período. Desea obtener también el total de cobros del período. Por cada cliente se ingresa el nombre, dirección, registro inicial y registro final. El cobro se efectúa con base en lo siguiente:

140 KVH o Menos	→	$ 35	
Los siguientes 170 KVH	→	$ 0.98	por KVH
Exceso sobre 310 KVH	→	$ 0.67	por KVH

Haga un diagrama de flujo para calcular lo solicitado anteriormente.

Datos: NOM_1, DIR_1, $REGINI_1$, $REGFIN_1$, NOM_2, DIR_2, $REGINI_2$, $REGFIN_2$,. . ., " bb ", " bb ", 0, 0

www.FreeLibros.com

Donde:

NOM_i	es una variable de tipo cadena de caracteres que representa el nombre de la persona i.
DIR_i	es una variable de tipo cadena de caracteres que representa la dirección de la persona i.
$REGFIN_i$	es una variable de tipo real que representa el registro final de la persona i.
$REGINI_i$	es una variable de tipo real que significa el registro inicial de la persona i.

Resultados:

Por cada cliente se debe imprimir un recibo. A continuación se presenta un ejemplo:

COMPAÑIA DE LUZ DEL CENTRO

.
.
.

Gustavo Nápoles
Expropiación Petrolera 328–5
Col. Aragón de los Frailes
01579-D.F.

www.FreeLibros.com

Donde:

NOM_i	es una variable de tipo cadena de caracteres que representa el nombre de la persona i.
DIR_i	es una variable de tipo cadena de caracteres que representa la dirección de la persona i.
$REGFIN_i$	es una variable de tipo real que representa el registro final de la persona i.
$REGINI_i$	es una variable de tipo real que significa el registro inicial de la persona i.

Resultados:

Por cada cliente se debe imprimir un recibo. A continuación se presenta un ejemplo:

```
COMPAÑIA DE LUZ DEL CENTRO
.
.
.
Gustavo Nápoles
Expropiación Petrolera 328–5
Col. Aragón de los Frailes
01579-D.F.

217 KVH
Costo : $110.46
```

Al final se debe imprimir también el total de cobros que tiene que hacer la compañía en el período.

PS 3.21

En una universidad se necesita obtener una lista de los alumnos candidatos a la Facultad Menor de Economía, Administración, Contabilidad e Ingeniería en Computación. Las condiciones para ser seleccionado como presunto candidato para entrar a la facultad menor, varían de carrera en carrera y se presentan a continuación:

Condiciones para Facultad Menor de Economía

Semestre $\geq$ 5 y Promedio $\geq$ 8.5

www.FreeLibros.com

Condiciones para Facultad Menor de Administración

Semestre ≥ 6 y Promedio ≥ 8.8

Condiciones para Facultad Menor de Contabilidad

Semestre ≥ 6 y Promedio ≥ 8.5

Condiciones para Facultad Menor de Computación

Semestre ≥ 6 y Promedio ≥ 8.8

Por cada uno de los N alumnos de la universidad se ingresa su matrícula, carrera en la que está inscrito, semestre que cursa y promedio general hasta el momento. Haga un diagrama de flujo para calcular lo solicitado anteriormente.

Datos: N, MAT_1, $CARR_1$, SEM_1, PRO_1, . . ., MAT_N, $CARR_N$, SEM_N, PRO_N

Donde:

N es una variable de tipo entero que representa el número de alumnos de la universidad.

MAT_i es una variable de tipo entero que representa la matrícula del alumno i ($1 \leq i \leq N$).

$CARR_i$ es una variable de tipo entero que representa la carrera en la que está inscrito el alumno i ($1 \leq i \leq N$). La carrera se ingresa teniendo en cuenta lo siguiente:

1 : " ECONOMIA "
2 : " ADMINISTRACION "
3 : " CONTABILIDAD "
4 : " INGENIERIA EN COMPUTACION "

SEM_i es una variable de tipo entero que representa el semestre que cursa el alumno i ($1 \leq i \leq N$).

PRO_i es una variable de tipo real que significa el promedio del alumno i ($1 \leq i \leq N$).

PS 3.22

En una clase de una universidad se tienen 35 alumnos. Haga un diagrama de flujo que calcule e imprima la matrícula y el promedio de calificaciones de cada alumno. Cabe aclarar que cada alumno de la clase tiene 5 calificaciones.

Datos: MAT_1, $CAL1_1$, $CAL2_1$, $CAL3_1$, $CAL4_1$, $CAL5_1$,. . ., MAT_{35}, $CAL1_{35}$, $CAL2_{35}$, $CAL3_{35}$, $CAL4_{35}$, $CAL5_{35}$

www.FreeLibros.com

Donde:

MAT_i	es una variable de tipo entero que representa la matrícula del alumno i ($1 \leq i \leq 35$).
$CAL1_i$, $CAL2_i$, $CAL3_i$, $CAL4_i$, $CAL5_i$	son variables de tipo real que representan las calificaciones del alumno i ($1 \leq i \leq 35$).

PS 3.23

Resuelva el ejercicio anterior, pero considere que las cinco calificaciones del alumno se leerán dentro de un ciclo.

Datos: MAT_1, $CAL_{1,1}$, $CAL_{1,2}$, . . ., $CAL_{1,5}$, . . . MAT_{35}, $CAL_{35,1}$,. . ., $CAL_{35,5}$

Donde:

MAT_i	es una variable de tipo entero que representa la matrícula del alumno i ($1 \leq i \leq 35$).
$CAL_{i,j}$	es una variable de tipo real que representa la calificación j del alumno i ($1 \leq j \leq 5$, $1 \leq i \leq 35$).

PS 3.24

Resuelva el problema anterior con las consideraciones planteadas en el mismo, pero sólo imprima matrícula y promedio del mejor y del peor alumno.

Datos: MAT_1, $CAL_{1,1}$, . . . $CAL_{1,5}$, . . . , MAT_{35}, $CAL_{35,1}$, $CAL_{35,5}$

PS 3.25

En una escuela en la que se tienen registros con las características físicas de los alumnos, se desea conocer la lista de los alumnos con aptitudes para practicar básquet. Haga un diagrama de flujo que obtenga lo siguiente:

a) Lista de alumnas con aptitudes físicas para jugar al básquet.

Requerimientos : ALTURA $\geq$ 1. 73 y 50 $\leq$ PESO $\leq$ 90.

b) Porcentaje de alumnas con estas aptitudes de la población estudiantil femenina.
c) Lista de alumnos con aptitudes físicas para jugar al básquet.

Requerimientos : ALTURA $\geq$ 1.83 y 73 $\leq$ PESO $\leq$ 110.

www.FreeLibros.com

d) Porcentaje de alumnos con estas aptitudes de la población estudiantil masculina.

Por cada alumno se ingresa su NOMBRE, SEXO, EDAD, PESO y ALTURA.

Datos: NOM_1, SEX_1, $EDAD_1$, $PESO_1$, ALT_1, NOM_2, SEX_2, $EDAD_2$, $PESO_2$, ALT_2, . . ., " X ", " X ", –1. –1,–

Donde:

NOM_i es una variable de tipo cadena de caracteres que representa el NOMBRE del alumno i.

SEX_i es una variable de tipo caracter que expresa el SEXO del alumno i. Se ingresa " F " para mujer y " M " para hombre.

EDADi es una variable de tipo entero que representa la EDAD del alumno i.

PESOi es una variable de tipo entero que representa el PESO del alumno i.

ALTi es una variable de tipo real que representa la ALTURA del alumno i.

PS 3.26

En una universidad con un número determinado de alumnos se desea obtener el porcentaje y promedio de la población femenina, el porcentaje y promedio de la población masculina y el promedio general. Por cada alumno se ingresa MATRICULA, SEXO, SEMESTRE y PROMEDIO. Haga un diagrama de flujo para calcular lo solicitado anteriormente.

Datos: N, MAT_1, SEX_1, SEM_1, PRO_1, . . ., MAT_N, SEX_N, SEM_N, PRO_N

Donde:

N es una variable de tipo entero que representa el número de alumnos.

MAT_i es una variable de tipo entero que representa la matrícula del alumno i ($1 \leq i \leq N$).

SEX_i es una variable de tipo entero que representa el sexo del alumno i ($1 \leq i \leq N$). Se ingresa 0 si es mujer y 1 si es hombre.

SEM_i es una variable de tipo entero que expresa el semestre del alumno i ($1 \leq i \leq N$).

PRO_i es una variable de tipo real que representa el promedio del alumno i ($1 \leq i \leq N$).

www.FreeLibros.com

PS 3.27

Dados los datos del problema anterior, haga un diagrama de flujo para calcular lo siguiente:

a) Matrícula y mayor promedio de la población femenina.
b) Matrícula y mayor promedio de la población masculina.
c) Promedios de los alumnos del 1er, 3er, 5to. y 7mo. semestre.

Datos: N, MAT_1, SEX_1, SEM_1, PRO_1, . . . , MAT_N, SEX_N, SEM_N, PRO_N

PS 3.28

En una empresa con N empleados se necesita obtener cierta información. Por cada empleado se ingresan los siguientes datos: CLAVE, EDAD, SEXO y SUELDO. Haga un diagrama de flujo para calcular e imprimir lo siguiente:

a) Número de hombres.
b) Número de mujeres.
c) Número de mujeres que ganen más de $20,000.
d) Número de hombres menores de 40 años que ganan menos de $40,000.
e) Número de empleados mayores de 50 años.

Datos: N, $CLAVE_1$, $EDAD_1$, $SEXO_1$, $SUELDO_1$,...,$CLAVE_N$, $EDAD_N$, $SEXO_N$, $SUELDO_N$

Donde:

N	es una variable de tipo entero que representa el número de empleados de la empresa.
$CLAVE_i$	es una variabe de tipo entero que representa la clave del empleado i ($1 \leq i \leq N$).
$EDAD_i$	es una variable de tipo entero que representa la edad del empleado i ($1 \leq i \leq N$).
$SEXO_i$	es una variable de tipo entero que representa el sexo del empleado i ($1 \leq i \leq N$). Se considera 1 si es hombre y 0 si es mujer.
$SUELDO_i$	es una variable de tipo real que representa el sueldo del empleado i ($1 \leq i \leq N$).

PS 3.29

La UNICEF desea obtener información estadística sobre los orfanatorios ubicados dentro de la República Mexicana y sobre los niños huérfanos internados en esos orfanatorios. Por cada niño se ingresa la siguiente información: nombre del

www.FreeLibros.com

huérfano, sexo, edad, nombre del orfanatorio y Estado de la República al que pertenece dicho orfanatorio.

Haga un diagrama de flujo para calcular lo siguiente:

a) Porcentaje de huérfanos del Estado de *México* y del *DF*, respecto al total del país.
b) Número de huérfanos en cada grupo. Los grupos se definen con base en la edad, teniendo en cuenta lo siguiente:

Grupo 1 : Edad menor a 1 año.
Grupo 2 : Edad comprendida entre 1 y 3 años (inclusive).
Grupo3 : Edad comprendida entre 4 y 6 años (inclusive).
Grupo4 : Edad mayor de 6 años.

c) Establecer el grupo que tiene la mayor cantidad de huérfanos.

Nota: Considere que el número de huérfanos en cada grupo es diferente.

Datos: N, NOM_1, SEX_1, $EDAD_1$, $NORF_1$, EST_1, ... , NOM_N, SEX_N, $EDAD_N$, $NORF_N$, EST_N

Donde:

N es una variable de tipo entero que representa el número de niños en los orfanatorios.

NOM_i es una variable de tipo cadena de caracteres que representa el nombre del niño i ($1 \leq i \leq N$).

SEX_i es una variable de tipo entero que representa el sexo del huérfano i. Toma valor de 1 si es hombre y de 0 si es mujer ($1 \leq i \leq N$).

$NORF_i$ es una variable de tipo cadena de caracteres que representa el nombre del orfanatorio donde se encuentra el huérfano i ($1 \leq i \leq N$).

EST_i es una variable de tipo cadena de caracteres que representa el nombre del Estado de la República donde se encuentra el orfanatorio en que vive el huérfano i ($1 \leq i \leq N$).

PS 3.30

Es posible demostrar que la suma de un número "suficiente" de términos de la serie:

www.FreeLibros.com

$$\text{Sen}(X) = X - \frac{X^3}{3!} + \frac{X^5}{5!} - \frac{X^7}{7!} + \cdots$$

Es un número considerablemente cercano a SEN(X), y que la diferencia entre SEN(X) y la suma antes mencionada se vuelve menor conforme se toman más y más términos. Escriba un diagrama de flujo tal que dada una X cualquiera calcule el SEN(X) utilizando la serie anterior, de tal modo que la diferencia entre la serie y un nuevo término sea menor o igual a 0.01. Imprima el número de términos requerido para obtener esta precisión.

Dato: X (variable de tipo entero que representa el número que se ingresa).

PS 3.31

Escriba un diagrama de flujo que obtenga e imprima todos los números "Primos Gemelos" comprendidos entre A y B (enteros positivos). Los primos gemelos son parejas de números primos con una diferencia entre sí de exactamente dos. Ejemplo: 3 y 5 son primos gemelos.

Datos: A, B (variables de tipo entero que representa los números enteros positivos que se ingresan. A tiene que ser menor que B).

PS 3.32

Una empresa automotriz necesita manejar los montos de ventas de sus N sucursales, en los últimos 14 años. Haga un diagrama de flujo que calcule lo siguiente:

a) Sucursal que más ha vendido en cada año.

b) Promedio de ventas por año.

c) Año con mayor promedio de ventas.

d) Ventas totales de la empresa (Considerando las N sucursales durante los 14 años).

Datos: N,
$Venta_{1,1}$, $Venta_{1,2}$,..., $Venta_{1,N}$
$Venta_{2,1}$, $Venta_{2,2}$, ..., $Venta_{2,N}$
$Venta_{14,1}$, $Venta_{14,2}$,... $Venta_{14,N}$

Donde:

N es una variable de tipo entero que representa el número de sucursales de la empresa ($1 \leq N \leq 50$).

www.FreeLibros.com

$VENTA_{i,j}$ es una variable de tipo real que representa el monto de ventas en el año i, de la sucursal j ($1 \leq i \leq 14, 1 \leq j \leq N$).

PS 3.33

El máximo común divisor (MCD) entre dos números es el natural más grande que divide a ambos. Construya un diagrama de flujo que calcule el máximo común divisor (MCD) de dos números naturales A y B.

Datos: A y B (variables de tipo entero que representan los números para calcular el MCD).

PS 3.34

Escriba un diagrama de flujo que imprima todos los pares de m y n que cumplan con la siguiente condición:

$$m^4 + 7*n^2 < 540$$

Nota: m y n sólo pueden ser enteros positivos.

PS 3.35

Escriba un diagrama de flujo que imprima todos los valores de X, Y y Z que satisfacen la siguiente expresión.

$$18*X^3 + 11*Y^5 + 8*Z^6 < 6300$$

Nota: X, Y y Z sólo pueden ser enteros positivos.

PS 3.36

En una granja llevan el registro de los kilogramos producidos y entregados mensualmente al mercado de la ciudad. Esta consiste de tomates, lechuga, acelgas, zanahorias y chícharos. Haga un diagrama de flujo que calcule lo siguiente:

a) ¿Cuál es el producto que más kilogramos rindió al final del año?
b) ¿Cuál fue la producción total de tomates, lechuga, acelgas, zanahorias y chícharos?, y Cuál fue el dinero que estos productos produjeron?
c) ¿Cuál es el producto que más dinero produjo al final del año?
d) Qué importe mensual le pagaron a esta familia de granjeros, por sus productos entregados al mercado.

www.FreeLibros.com

Datos: TOM_1, $PTOM_1$, LEC_1, $PLEC_1$, ACE_1, $PACE_1$, ZAN_1, $PZAN_1$, CHI_1, $PCHI_1$,
TOM_2, $PTOM_2$, LEC_2, $PLEC_2$, ACE_2, $PACE_2$, ZAN_2, $PZAN_2$, CHI_2, $PCHI_2$,
.....
TOM_{12}, $PTOM_{12}$, LEC_{12}, $PLEC_{12}$, ACE_{12}, $PACE_{12}$, ZAN_{12}, $PZAN_{12}$, CHI_{12}, $PCHI_{12}$

Donde:

TOM_i es una variable de tipo real que representa la cantidad de tomates (en kilogramos) entregados al mercado en el mes i ($1 \leq i \leq 12$).

$PTOM_i$ es una variable de tipo real que expresa el precio del kilo de tomate en el mes i ($1 \leq i \leq 12$).

LEC_i es una variable de tipo real que representa la cantidad de lechuga entregadas al mercado en el mes i ($1 \leq i \leq 12$).

$PLEC_i$ es una variable de tipo real que expresa el precio del kilo de lechuga en el mes i ($1 \leq i \leq 12$).

ACE_i es una variable de tipo real que representa los kilogramos de acelga entregada al mercado en el mes i ($1 \leq i \leq 12$).

$PACE_i$ es una variable de tipo real que expresa el precio del kilo de acelgas en el mes i ($1 \leq i \leq 12$).

ZAN_i es una variable de tipo real que representa los kilogramos de zanahoria entregada al mercado en el mes i ($1 \leq i \leq 12$).

$PZAN_i$ es un variable de tipo real que expresa el precio del kilogramo de zanahoria en el mes i ($1 \leq i \leq 12$).

CHI_i es una variable de tipo real que representa los kilogramos de chícharos entregados al mercado en el mes i ($1 \leq i \leq 12$).

$PCHI_i$ es una variable de tipo real que expresa el precio del kilogramo de chícharos en el mes i ($1 \leq i \leq 12$).

Nota: Puede ocurrir que en un determinado mes no se entregue un cierto producto al mercado, en ese caso la producción de ese producto se ingresará como 0. Tomate en algunos países de habla hispana es conocido con el nombre de jitomate. Chícharos es conocido con el nombre de arvejas.

PS 3.37

En una tienda se venden diferentes tipos de productos y cada uno de éstos tiene una clave asociada al ramo al que pertenecen.

www.FreeLibros.com

CLAVE	*RAMO*
1	Perecederos
2	Aseo
3	Perfumería
4	Abarrotes

Haga un diagrama de flujo que realice lo siguiente.

a) Confeccione una factura por cliente. Cada factura debe tener el siguiente formato:

TIENDA "LA PROVIDENCIA"
MEXICO, D.F.

CLAVE -RAMO	*CANTIDAD*	*PRECIO UNITARIO*	*TOTAL*
XX	YY	ZZ	WW
XX	YY	ZZ	WW

TOTAL A PAGAR : $ GGG

b) Al final del día imprima el total vendido en $ de cada uno de estos 4 ramos.
c) Imprima además el ramo (teniendo en cuenta el dinero producido) que más se consume en esta tienda.

Datos: N_1
$CLAVE_1$, $CANTIDAD_1$, $PRECIO_UNIT_1$
$CLAVE_2$, $CANTIDAD_2$, $PRECIO_UNIT_2$
. . . . . .
$CLAVE_{N1}$, $CANTIDAD_{N1}$, $PRECIO_UNIT_{N1}$

N_2
$CLAVE_1$, $CANTIDAD_1$, $PRECIO_UNIT_1$
$CLAVE_2$, $CANTIDAD_2$, $PRECIO_UNIT_2$
. . . . . .
$CLAVE_{N2}$, $CANTIDAD_{N2}$, $PRECIO_UNIT_{N2}$

...
-1

Donde:

N_i es una variable de tipo entero que representa el número de productos que compró el cliente i.

www.FreeLibros.com

CLAVEj es una variable de tipo entero que representa la clave del producto que compró el cliente i del producto con clave j ($1 \leq j \leq 4$).

CANTIDADj es una variable de tipo entero que representa la cantidad de productos que compró el cliente i del producto con clave j ($1 \leq j \leq 4$).

PRECIO_UNITj es una variable de tipo real que representa el precio del producto con clave j que compró el cliente i ($1 \leq j \leq 4$).

PS 3.38

En un restaurante se sirven 7 platillos diferentes. Cada platillo se reconoce por una clave, que es un valor numérico comprendido entre 1 y 7. Diariamente se atienden a numerosos clientes, por lo que es necesario generar facturas con este formato:

FACTURA NUMERO : 1

CLAVE	*CANTIDAD*	*PRECIO UNITARIO*	*PRECIO TOTAL*
5	1	45	45
6	2	30	60
9	1	38	38

TOTAL A PAGAR : ——————————————————————————-$ 143

Escriba un diagrama de flujo que haga lo siguiente:

a) Lea los precios de los 7 platillos.

b) Lea los datos de cada cliente y entregue una factura como la presentada anteriormente. Las facturas se enumeran de 1 hasta N, desde el comienzo del día.

c) Calcule lo que ha vendido el restaurante al final del día.

Datos: P1, P2, P3, P4, P5, P6, P7

K_1,$CLAVE_1$,$CANT_1$,$CLAVE_2$,$CANT_2$,...,$CLAVE_{k1}$,$CANT_{k1}$
K_2,$CLAVE_1$,$CANT_1$,$CLAVE_2$,$CANT_2$,...,$CLAVE_{k2}$,$CANT_{k2}$
. . .
-1

www.FreeLibros.com

Donde:

P1, P2, P3, P4, P5, P6 y P7: son variables de tipo real que representan los precios de los 7 platillos.

K_i es una variable de tipo entero que representa el número de platillos que ha consumido el cliente i.

$CLAVE_j$ es una variable de tipo entero que representa la clave del platillo j consumido por el cliente i ($1 \leq j \leq 7$).

$CANT_j$ es una variable de tipo entero que representa la cantidad de platillos de la clave j consumidos por el cliente i ($1 \leq j \leq 7$).

PS 3.39

Escriba un diagrama de flujo que calcule F(X,Y,Z) para cada uno de los datos que se proporcionan.

$$F(X, Y, Z) = \begin{cases} \prod_{i=1}^{X} & \text{si } X > 0 \\ \frac{Z!}{Y!} & \text{si } X > 0, Z > 0 \text{ e } Y > 0 \\ 1 & \text{En cualquier otro caso} \end{cases}$$

Datos: $K, X_1, Y_1, Z_1, X_2, Y_2, Z_2, \ldots, X_K, Y_K, Z_K$

Donde:

K es una variable de tipo entero que representa el número de veces que se calculará la función.

X, Y, y Z son variables de tipo entero que representan los datos que se ingresan.

PS 3.40

Se tiene un grupo de registros donde se almacena información relativa a los elementos de la tabla periódica de química. Por cada elemento se ingresa su nombre, su conductividad eléctrica y su conductividad térmica. Haga un diagrama de flujo para calcular lo siguiente :

a) Los dos principales elementos conductores de la electricidad y el calor.

b) Los dos peores elementos conductores de la electricidad y el calor.

Datos: $NOM_1, CE_1, CT_1, NOM_2, CE_2, CT_2, \ldots$, "NN",0,0

www.FreeLibros.com

Donde:

NOM_i es una variable de tipo cadena de caracteres que representa el nombre del elemento i.

CE_i es una variable de tipo real que representa la conductividad eléctrica del elemento i. Si no se conoce el valor del elemento i se ingresa como CE el valor de -1.

CT_i es una variable de tipo real que representa la conductividad térmica del elemento i. Si no se conoce el valor del elemento i se ingresa como CT el valor de -1.

El fin de datos está expresado como : "NN",0,0.

PS 3.41

En una Universidad se conoce el número de alumnos que ingresaron en sus 5 diferentes carreras, en los últimos 10 años. Haga un diagrama de flujo que proporcione la siguiente información:

a) Total de alumnos por año que ingresaron a la Universidad.
b) Porcentaje de alumnos ingresados en el año X de la carrera Y.
c) En qué año y en qué carrera se dio el menor ingreso de alumnos.
d) Año en el cual la carrera T tuvo el mayor ingreso de alumnos.

Datos: AÑOX,CARR_Y,CARR_T
$ALUM_{1,1}$,$ALUM_{1,2}$,...,$ALUM_{1,5}$
$ALUM_{2,1}$,$ALUM_{2,2}$,...,$ALUM_{2,5}$
. . .
. . .
. . .
$ALUM_{10,1}$,$ALUM_{10,2}$,...,$ALUM_{10,5}$

Donde:

$ALUM_{i,j}$ es una variable de tipo entero que representa el número de alumnos que ingresaron a la carrera j en el año i ($1 \leq i \leq 10, 1 \leq 5 \leq j$).

AÑOX es una variable de tipo entero que representa el año en el cual se quiere obtener el porcentaje para contestar el punto b.

CARR_Y es una variable de tipo entero que representa la carrera de la cual se quiere obtener el porcentaje para contestar el punto b.

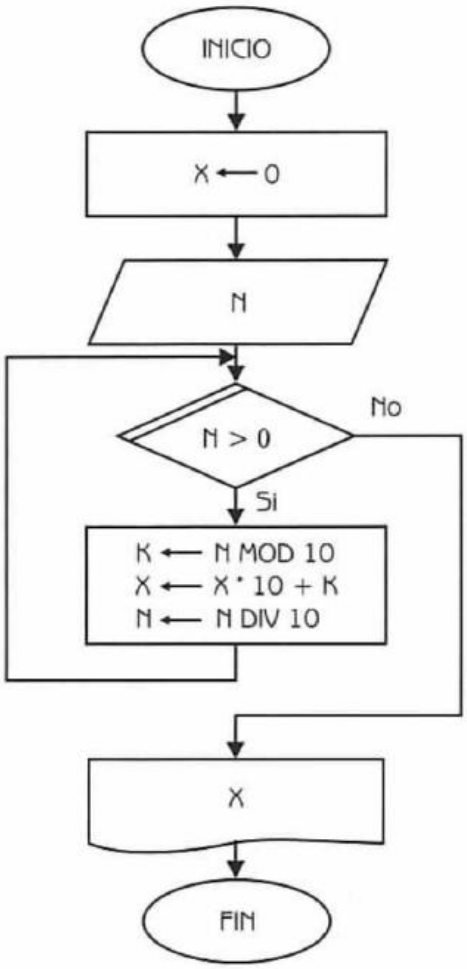
CARR_T es una variable de tipo entero que representa la carrera del punto d, de la cual se quiere obtener el año que tuvo el mayor ingreso de alumnos.

www.FreeLibros.com

PS 3.42

El siguiente diagrama de flujo tiene un fin específico. Realice lo siguiente:

a) Interprételo. Diga qué hace en forma breve.
b) Sígalo para N = 20 y N = 484.
c) Marque en el diagrama de flujo las estructuras algorítmicas utilizadas.



Nota:
X, N y K son variables de tipo entero.

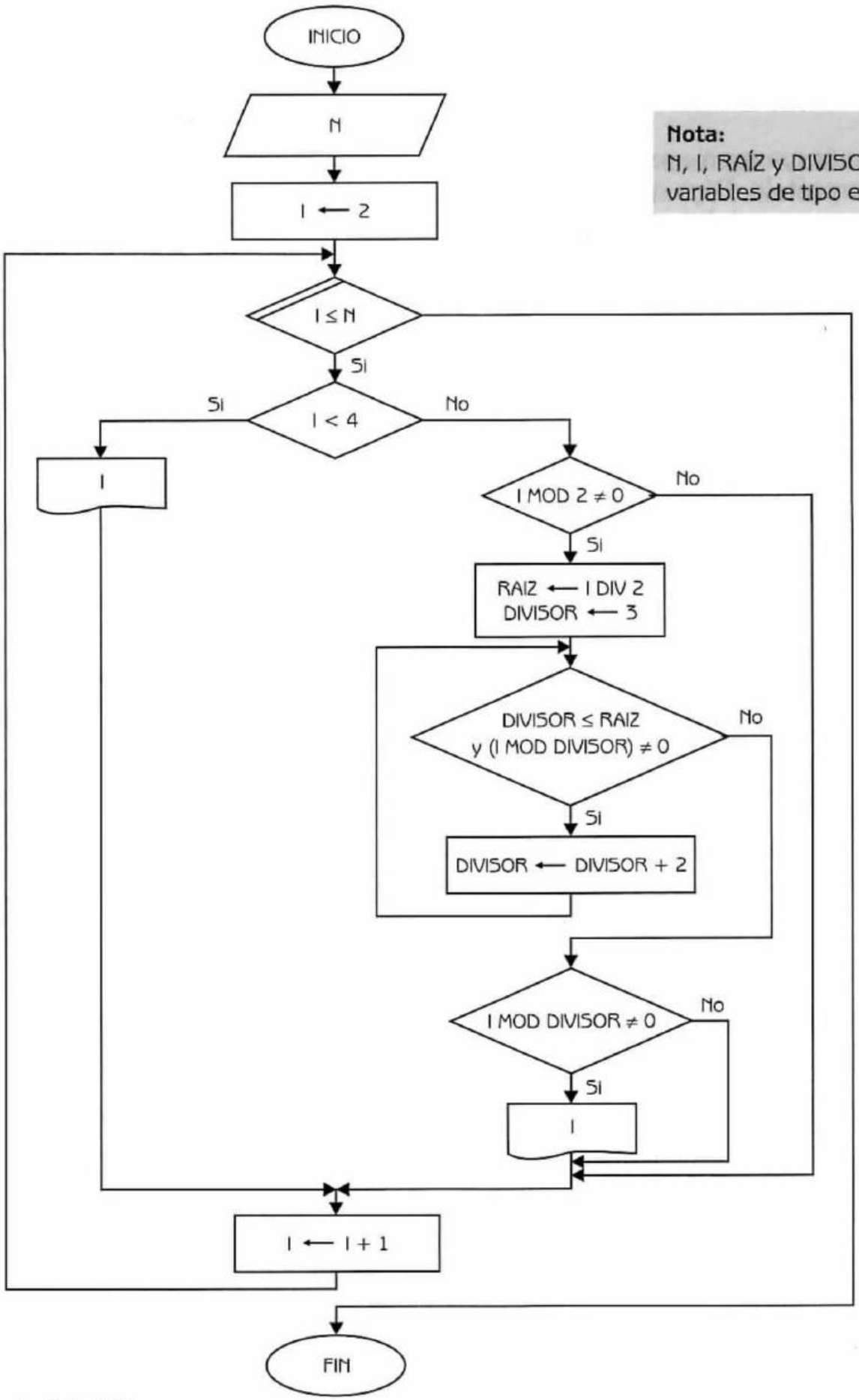
Diagrama de flujo 6.1

PS 3.43

El siguiente diagrama de flujo fue construido con un fin específico. Realice lo siguiente:

a) Construya un mapa de memoria para N = 11, indicando lo que imprime el programa.
b) Interprételo. Diga qué hace en forma breve.
c) Marque en el diagrama de flujo las estructuras algorítmicas utilizadas.

www.FreeLibros.com



Nota:
N, I, RAÍZ y DIVISOR son variables de tipo entero.

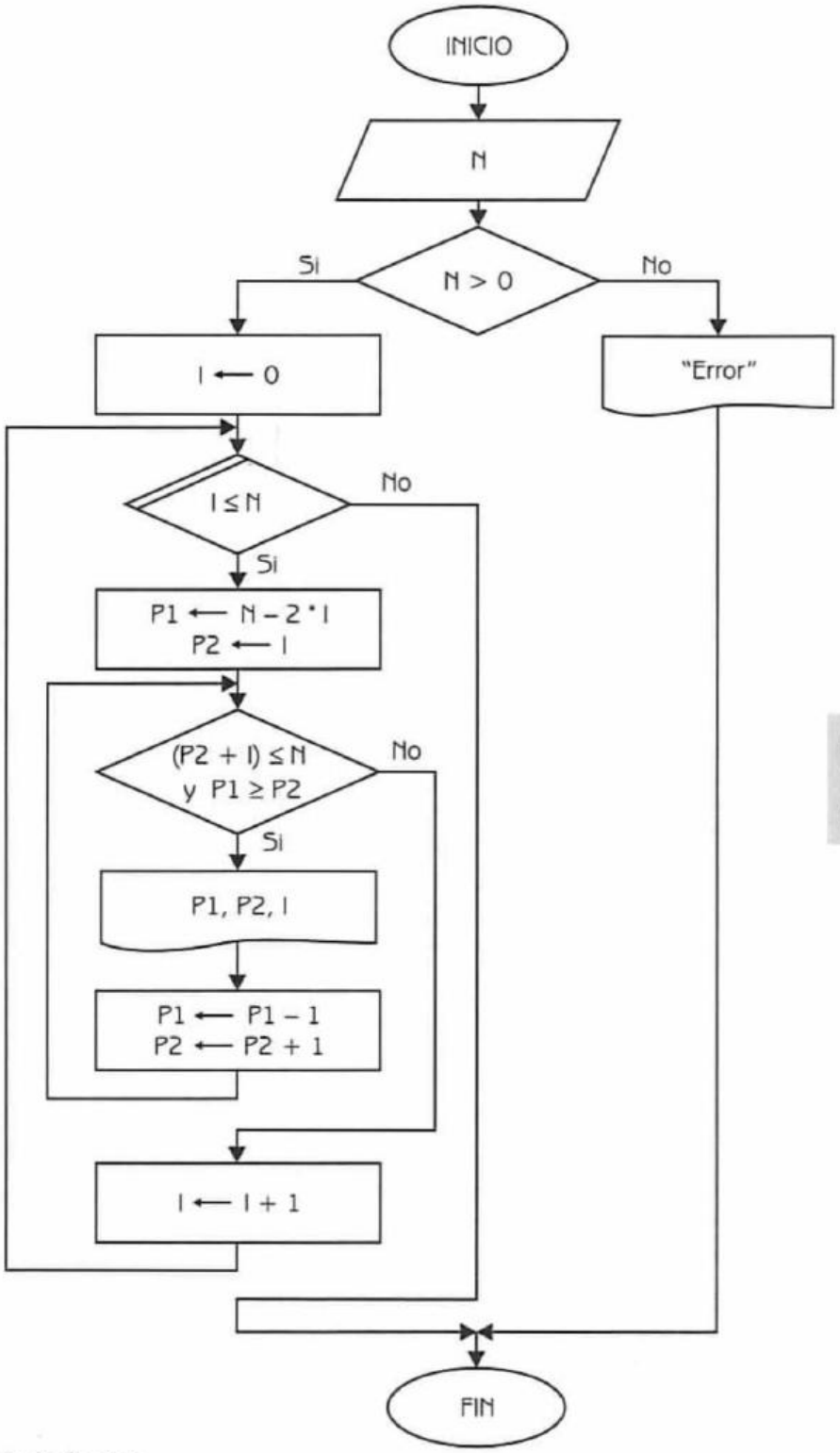
Diagrama de flujo 6.2

www.FreeLibros.com

PS 3.44

El siguiente diagrama de flujo fue construido con un fin específico. Realice lo siguiente:

a) Construya un mapa de memoria para N = 5, indicando lo que se imprime.
b) Interprételo. Diga qué hace en forma breve.
c) Marque en el diagrama de flujo las estructuras algorítmicas utilizadas.



Nota:
N, I, P1 y P2 son variables de tipo entero.

Diagrama de flujo 6.3

www.FreeLibros.com

Arreglos Unidimensionales

PS 4.1

Escriba un diagrama de flujo tal que dado como entrada un arreglo unidimensional de números reales, obtenga como resultado la suma de los mismos.

Dato: VEC [1..100] (arreglo unidimensional de números reales).

PS 4.2

Construya un programa tal que dado como entrada un arreglo unidimensional de enteros y un número entero, determine cuántas veces se encuentra el número dentro del arreglo.

Datos: VEC [1..100], NUM

Donde: VEC es un arreglo unidimensional de tipo entero de 100 elementos.
NUM es una variable de tipo entero.

PS 4.3

Escriba un diagrama de flujo tal que dado como entrada un arreglo unidimensional que contiene números enteros, determine cuántos de ellos son positivos, negativos o nulos.

Dato: ARRE [1..300] (arreglo unidimensional de tipo entero).

PS 4.4

Dados dos vectores de tipo entero A y B, construya un diagrama de flujo que calcule el producto de dichos vectores.

Datos: A[1..N], B[1..N] (arreglos unidimensionales de tipo entero, $1 \leq N \leq 50$).

PS 4.5

Se tienen registradas en un arreglo unidimensional ALU, las calificaciones obtenidas en un examen por un grupo de 250 alumnos. Cada calificación es un número entero comprendido entre 0 y 10.

www.FreeLibros.com

Escriba un diagrama de flujo que calcule e imprima la frecuencia de cada calificación. La salida del programa debe ser como la que se muestra a continuación:

CALIFICACION	FRECUENCIA
0	5
1	6
2	15
.	.
.	.
.	.
10	20

Dato: ALU [1..250] (arreglo unidimensional de tipo entero que almacena las calificaciones de un grupo de 250 estudiantes).

PS 4.6

Construya un diagrama de flujo para almacenar en un arreglo unidimensional los 100 primeros números Fibonacci. Imprima al final el arreglo correspondiente.

PS 4.7

La búsqueda secuencial consiste en revisar elemento por elemento hasta encontrar el dato buscado, o hasta llegar al final de la lista de datos disponibles. Cuando se habla de búsqueda en arreglos debe distinguirse entre arreglos desordenados y arreglos ordenados.

La búsqueda secuencial en arreglos desordenados consiste básicamente en recorrer el arreglo de izquierda a derecha hasta que se encuentre el elemento buscado o se termine el arreglo, lo que ocurra primero. Normalmente cuando un procedimiento de búsqueda concluye con éxito interesa conocer en qué posición fue hallado el elemento buscado. Esta idea puede generalizarse para todos los métodos de búsqueda. Construya el diagrama de flujo correspondiente.

Datos: VECTOR [1..N] , X $\qquad 1 \leq N \leq 50$

Donde:

VECTOR es un arreglo unidimensional de tipo entero que se encuentra desordenado.

X es una variable de tipo entero que representa el elemento que buscamos en el arreglo.

www.FreeLibros.com

PS 4.8

La búsqueda secuencial en arreglos ordenados es similar al problema de búsqueda secuencial en arreglos desordenados. Sin embargo como el arreglo está ordenado, se impone una nueva condición para controlar la búsqueda. Cuando el valor buscado no esté en el arreglo y sea un valor intermedio, V[I] < X < V[N], se detendrá la búsqueda al dejar de cumplirse la condición X ≥ V[I]. Si X no se encontró hasta el momento, tampoco se encontrará entre los restantes elementos. Construya el diagrama de flujo correspondiente.

Datos: VECTOR [1..N], X $1 \leq N \leq 50$

Donde:

VECTOR es un arreglo unidimensional de tipo entero que se encuentra ordenado en forma ascendente.

X es una variable de tipo entero que representa el elemento a buscar en el arreglo.

PS 4.9

Construya un diagrama de flujo que permita insertar y eliminar elementos en un arreglo unidimensional que se encuentra desordenado.

Dato: ARRE[1..N] (arreglo unidimensional de tipo entero, $1 \leq N \leq 100$).

PS 4.10

Construya un diagrama de flujo que permita insertar y eliminar elementos en un arreglo unidimensional de tipo entero que se encuentra ordenado.

Dato: ARRE [1..N] (arreglo unidimensional de tipo entero, $1 \leq N \leq 100$).

PS 4.11

Escriba un diagrama de flujo tal que dado un arreglo unidimensional de tipo cadena de caracteres que contiene los nombres de las ciudades de más de 50 000 habitantes de un país, determine dado el nombre de una ciudad, si ésta se encuentra en el arreglo.

Datos: CIUDADES [1..N], CIU $1 \leq N \leq 1000$

www.FreeLibros.com

Donde:

CIUDADES es un arreglo unidimensional de tipo cadena de caracteres que contiene las ciudades del país con más de 50 000 habitantes.

CIU es una variable de tipo cadena de caracteres que representa la ciudad que se ingresa.

Nota: El arreglo CIUDADES no se encuentra ordenado.

PS 4.12

En una papelería se maneja información sobre los 8 modelos diferentes de cuadernos que venden. Se conocen los precios de cada modelo:

$$P_1, P_2, P_3, \ldots, P_8$$

Donde:

P_i es una variable de tipo real que representa el precio del modelo i.

Por otra parte se tiene información sobre las ventas realizadas durante los últimos 30 días. Los datos se presentan de esta forma:

$$DIA_1, MOD_1, CANT_1$$
$$DIA_2, MOD_2, CANT_2$$
$$\ldots$$
$$-1, -1, -1$$

Donde:

DIA_i es una variable de tipo entero que representa el día en que se efectuó la venta i ($1 \leq DIA \leq 30$).

MOD_i es una variable de tipo entero que representa el modelo que se vendió en la venta i ($1 \leq MODELO \leq 8$).

$CANT_i$ es una variable de tipo entero que representa el número de cuadernos que se vendió en la venta i.

Construya un diagrama de flujo que calcule lo siguiente:

a) El total recaudado por modelo en los 30 días.

b) El total recaudado por día.

c) ¿Cuál fue el modelo que más dinero produjo en los 30 días?

www.FreeLibros.com

PS 4.13

Se tienen dos arreglos: CINES y TEATROS. El primero almacena los nombres de todos los cines de la ciudad. Está ordenado alfabéticamente de manera ascendente. El segundo arreglo guarda los nombres de todos los teatros de la ciudad y está ordenado alfabéticamente de manera descendente.

Construya un diagrama de flujo que mezcle estos arreglos formando un tercero, ENTRETENIMIENTOS, de tal manera que quede alfabéticamente de manera ascendente.

Datos: CINES[1..N], TEATROS[1..M] $(1 \leq N \leq 100,\ 1 \leq M \leq 50)$.

Donde:

CINES	es un arreglo unidimensional de tipo cadena de caracteres que almacena los nombres de los cines de la ciudad y está ordenado de manera ascendente.
TEATROS	es un arreglo unidimensional de tipo cadena de caracteres que almacena los nombres de los teatros de la ciudad y está ordenado alfabéticamente de manera descendente.

PS 4.14

Construya un diagrama de flujo que almacene en un arreglo unidimensional todos los años bisiestos y perfectos comprendidos desde el año 1 hasta el año 2020. Cabe aclarar que un año es bisiesto si es divisible entre 4, sin embargo no se considera bisiesto si es divisible entre 100 a menos que sea divisible entre 400.
Por otra parte, un número se considera perfecto si es igual a la suma de sus divisores, incluyendo al 1 y exceptuando a él mismo. Por ejemplo, el 6 es perfecto (6 = 1 + 2 + 3).

www.FreeLibros.com

Arreglos bidimensionales

PS 4.15

Sean dos arreglos bidimensionales A(M × N) y B(M × N). Escriba un diagrama de flujo que calcule la suma de los arreglos A y B, y almacene el resultado en el arreglo bidimensional C(M × N).

Datos: A[1..M,1..N], B[1..M,1..N] (A y B arreglos bidimensionales de tipo real $1 \leq M \leq 10$, $1 \leq N \leq 20$).

PS 4.16

Escriba un diagrama de flujo tal que dado como dato una matriz A(N × N), escriba la diagonal de esa matriz.

Dato: A[1..N,1..N] (arreglo bidimensional de tipo real, $1 \leq N \leq 50$).

PS 4.17

Construya un diagrama de flujo que permita calcular la tabla de multiplicar del 1 al 10. Imprima al final el resultado en forma de tabla.

PS 4.18

Construya un diagrama de flujo que llene de ceros una matriz cuadrada A [N × N], excepto en la diagonal principal donde debe asignar 1. Si N es igual a 4, la matriz debe quedar de la siguiente manera:

	1	2	3	4
1	1	0	0	0
2	0	1	0	0
3	0	0	1	0
4	0	0	0	1

Dato: A[1..N,1..N] (arreglo bidimensional de tipo entero, $1 \leq N \leq 50$).

PS 4.19

Se tiene información sobre las calificaciones de 6 exámenes de un grupo de 30 alumnos. Los datos sobre estos exámenes se proporcionan de la siguiente manera:

www.FreeLibros.com

$$\begin{array}{l} CAL_{1,1}\ CAL_{1,2}\ \ldots\ldots\ CAL_{1,6} \\ CAL_{2,1}\ CAL_{2,2}\ \ldots\ldots\ CAL_{2,6} \\ \ldots\ldots\ldots\ldots \\ CAL_{30,1}\ CAL_{30,2}\ \ldots\ldots CAL_{30,6} \end{array}$$

Donde: CALi,j es una variable de tipo real que representa la calificación que obtuvo el alumno i en el examen j ($1 \leq i \leq 30$, $1 \leq j \leq 6$).

Escriba un diagrama de flujo que permita calcular lo siguiente:

a) El promedio de calificaciones de cada uno de los 6 exámenes.
b) El promedio de cada alumno.
c) El tipo (número) de examen que tuvo el mayor promedio de calificación. Escriba también dicho promedio.

PS 4.20

Sean A(M × N) y B(N) arreglos de dos y una dimensión, respectivamente. Construya un diagrama de flujo que asigne valores a A, a partir de B teniendo en cuenta los siguientes criterios:

a) A[I,J] = B[I] Si I ≤ J
b) A[I,J] = 0 Si I > J

Datos: A[1..M, 1..N], B[1..N] ($1 \leq M \leq 30$, $1 \leq N \leq 20$)

Donde:

A es un arreglo bidimensional de tipo real de M renglones y N columnas.
B es un arreglo unidimensional de N elementos de tipo real.

PS 4.21

En el arreglo bidimensional TEMP de 12 renglones × 31 columnas se almacenaron las temperaturas promedio diarias del año anterior, en la ciudad de México. Construya un diagrama de flujo que permita calcular lo siguiente:

a) La temperatura más alta registrada el año anterior, y cuál fue el día y mes en que se registró.
b) El mes que tuvo el promedio de temperaturas más alto.
c) EL promedio mensual de temperaturas.

Dato: TEMP[1..12,1..31] (arreglo bidimensional de tipo real).

www.FreeLibros.com

Nota: Recuerde que los meses 4, 6, 9 y 11 tienen 30 días. El mes 2 tuvo el año anterior 28 días.

PS 4.22

Los datos reunidos en la Secretaría de Industrias relacionado a la producción de N fábricas en cada uno de los meses del año anterior, se proporcionan de la siguiente manera:

Datos: N
FAB_1, $MES_{1,1}$, $MES_{1,2}$, ..., $MES_{1,12}$,
FAB_2, $MES_{2,1}$, $MES_{2,2}$, ..., $MES_{2,12}$,
...
...
FAB_N, $MES_{N,1}$, $MES_{N,2}$, ..., $MES_{N,12}$

Donde:

N es una variable de tipo entero que representa el número de fábricas, $1 \leq N \leq 500$.
FAB_i es una variable de tipo entero que representa la clave de la $fábrica_i$
MESi,j es una variable de tipo real que representa la producción de la $fábrica_i$ en el mes j.

Realice un diagrama de flujo que proporcione la siguiente información:

a) La clave de la fábrica que más produjo el año anterior. Imprimir también su producción.
b) Dado un mes como parámetro, imprimir las claves de las fábricas cuyas producciones en dicho mes, fueron superiores a $150 000.

PS 4.23

En una universidad se conoce el número de alumnos que ingresaron en sus 7 diferentes carreras, en los últimos 12 años. Construya un diagrama de flujo que pueda proporcionar la siguiente información:

a) Total de alumnos por año.
b) Porcentaje de alumnos ingresados en el año X de la carrera Y.
c) En qué año y en qué carrera se dio el menor ingreso de alumnos.
d) Año en el cual la carrera T tuvo el mayor ingreso de alumnos.

www.FreeLibros.com

Dato: ALUM[1..12,1..7] (arreglo bidimensional de tipo entero que almacena el número de alumnos que ingresaron en los últimos 12 años, en las 7 carreras de la universidad).

PS 4.24

Se tiene una tabla con las calificaciones obtenidas por un grupo de N alumnos en 5 exámenes diferentes. Construya un diagrama de flujo que permita calcular lo siguiente:

a) El promedio de calificaciones de cada uno de los N alumnos (considerando los 5 exámenes)
b) El alumno o los alumnos que obtuvieron la mejor calificación en el tercer examen.
c) El o los alumnos, si existieran, que obtuvieron la mayor calificación en el primero y en el quinto examen.
d) Dado un número que identifica a un alumno, informar en qué examen logró la menor calificación.
e) ¿En qué examen fue más alto el promedio de los N alumnos?

Dato: ALUM [1..N, 1..5] (arreglo bidimensional de tipo real que almacena las calificaciones de un grupo de N alumnos en 5 exámenes diferentes, $1 \leq N \leq 60$).

PS 4.25

En una fábrica se proporcionan los datos de producción de 10 diferentes productos en los 12 meses del año.

MES_1, $TIPO_1$, $CANT_1$
MES_2, $TIPO_2$, $CANT_2$
. . .

−1, −1, −1

Donde:

MES_i es una variable de tipo entero que hace referencia al mes de producción del producto i ($1 \leq MES \leq 12$).
$TIPO_i$ es una variable de tipo entero que representa el tipo de producto en la transacción i ($1 \leq MODELO \leq 8$).
$CANT_i$ es una variable de tipo entero que representa la cantidad producida de un producto en la transacción i.

www.FreeLibros.com

Además se conocen los costos de producción de cada uno de los productos:

$$COSTO_1, COSTO_2, \ldots, COSTO_{10}$$

Donde:

COSTO es una variable real que representa el costo de producción del producto i

Construya un diagrama de flujo que conteste lo siguiente:

a) El costo anual de producción de cada uno de los productos.

b) El costo total de producción de la fábrica.

c) Obtenga el tipo de producto que más se vendió en el año y el tipo de producto con mayor costo de producción anual.

PS 4.26

Construya un diagrama de flujo tal que dado como dato un arreglo bidimensional A(N × N), genere un arreglo unidimensional B(N) considerando las siguientes reglas:

$$B[i] = \begin{cases} B[i-2]/N & \text{Si } (i \bmod 4) = 0 \\ \prod_{j=1}^{N} A[i,j] & \text{Si } (i \bmod 4) = 1 \\ \sum_{j=1}^{N} A[i,j] & \text{Si } (i \bmod 4) = 2 \\ B[i-2]*B[i-1] & \text{Si } (i \bmod 4) = 3 \end{cases}$$

Imprima al final el arreglo B.

Datos: A[1..N,1..N] B[1..N] $1 \leq N \leq 30$

Donde:

A es un arreglo bidimensional cuadrado de tipo entero.
B es un arreglo unidimensional de tipo real.

PS 4.27

Se tiene una matriz A cuadrada de orden N. Se desea generar un vector de N elementos teniendo en cuenta el siguiente criterio:

www.FreeLibros.com

$$B[i] = \begin{cases} \sum_{k=1}^{i} A[i,k] & \text{Si } (i \bmod 3) = 1 \\ \prod_{k=i}^{n} A[k,i] & \text{Si } (i \bmod 3) = 2 \\ \left(\prod_{k=i-1}^{n} A[k,i-1]\right) / \left(\sum_{k=1}^{i-2} a[i-2,k]\right) & \text{De otra forma} \end{cases}$$

Construya un diagrama de flujo que realice lo siguiente:

a) Lea la matriz A.
b) Genere el vector B.
c) Escriba el vector B.

Datos: A[1..N, 1..N] (matriz cuadrada de tipo entero, $1 \leq N \leq 50$).

PS 4.28

Dada una matriz cuadrada MAT (M × N), construya un diagrama de flujo que determine si la misma puede considerarse como un cuadrado mágico. Un cuadrado mágico es aquel en que la suma de las filas, columnas y diagonales siempre tiene el mismo valor.

Dato: MAT[1..N, 1..N] (arreglo bidimensional de tipo entero, $1 \leq N \leq 50$).

PS 4.29

La administración de una compañía quiere conocer el ausentismo registrado en cada uno de los 8 departamentos que tiene la empresa en un mes laboral. Para ello se requiere elaborar un diagrama de flujo en el que se dan como entrada el número diario de ausentes en cada departamento. Como salida, se debe presentar una tabla mostrando el ausentismo diario de cada departamento y el departamento con mayor y menor ausentismo.

Datos: DIA, $AU_{1,1}$, $AU_{1,2}$, ..., $AU_{1,8}$
$AU_{2,1}$, $AU_{2,2}$, ..., $AU_{2,8}$
...
$AU_{DIA,1}$, $AU_{DIA,2}$, ..., $AU_{DIA,8}$

Donde:

DIA es una variable de tipo entero que representa el número de días hábiles en el mes laboral.

www.FreeLibros.com

$AU_{i,j}$ es una variable de tipo entero que representa el número de ausentes registrados el día i en el departamento j.

PS 4.30

Escriba un diagrama de flujo que permita manejar la venta de localidades de un teatro, en donde las filas están marcadas de la A a la Z y tiene asientos numerados del 1 al 10.

Se debe pedir al usuario la elección de asientos concretos hasta que escoja alguno que no esté reservado; posteriormente pedirá el nombre del cliente para asignarle su lugar.

Dato: TEATRO['A'..'Z', 1..10] (arreglo bidimensional de tipo cadena de caracteres que se utiliza para almacenar los nombres de las personas que hacen reservaciones para asistir a un teatro).

PS 4.31

Sean dos arreglos bidimensionales A(M × N) y B(N × P). Escriba un diagrama de flujo que calcule el producto de los dos arreglos A y B, y almacene el resultado en el arreglo bidimensional C(M × P).

Datos: A[1..M, 1..N], B[1..N, 1..P] (arreglos bidimensionales de tipo real, $1 \leq M \leq 10, 1 \leq N \leq 20, 1 \leq P \leq 8$).

PS 4.32

Dada una matriz A (M × N) y una matriz B (N × M), construya un diagrama de flujo que obtenga la suma de A más la traspuesta de B ($A + B^T$). Imprima la matriz resultante.

Datos: A[1..M, 1..N], B[1..N, 1..M] (arreglos bidimensionales de tipo entero, $1 \leq M \leq 10, 1 \leq N \leq 10$).

www.FreeLibros.com

Arreglos de más de dos dimensiones

PS 4.33

En una universidad se almacena información sobre el número de alumnos que han ingresado a sus diferentes carreras (7), en los dos semestres (enero/agosto), en los últimos 10 años. Para esto se almacena la información en un arreglo tridimensional. Construya un diagrama de flujo que proporcione la siguiente información:

a) El año en que ingresó el mayor número de alumnos a la universidad.
b) ¿Qué carrera en el último año recibió el mayor número de alumnos?
c) ¿En qué año la carrera de Economía recibió el mayor número de alumnos?
d) ¿En qué semestre en el año 7, la carrera de Ingeniería en Computación, recibió el mayor número de alumnos?

Datos: UNIV [1..7,1..2,1..10] (arreglo tridimensional de tipo entero que almacena información respecto al ingreso de alumnos a diferentes carreras en una universidad).

Consideraciones:

Tome en cuenta que el nombre de cada carrera está asociado a un valor numérico, como se representa a continuación:

1: Contabilidad
2: Economía
3: Administración
4: Ingeniería en Computación
5: Ingeniería en Telemática
6: Relaciones Internacionales
7: Actuaría

PS 4.34

Dado un arreglo tridimensional que contiene la producción correspondiente a los 12 meses de los últimos 4 años, de los departamentos de dulces, conservas y bebidas de una fábrica; construya un diagrama de flujo que proporcione la siguiente información:

a) ¿Cuál fue el departamento que más producción obtuvo al cabo de los 4 años?
b) ¿En qué mes del segundo año, se alcanzó el mayor nivel de producción en la fábrica?

www.FreeLibros.com

c) ¿En qué año se produjo más?
d) ¿En qué departamento, mes y año se tuvo el mayor nivel de producción de la compañía?

Datos: FABRICA [1..12, 1..4, 1..3] (arreglo tridimensional de tipo real).

Arreglos paralelos

PS 4.35

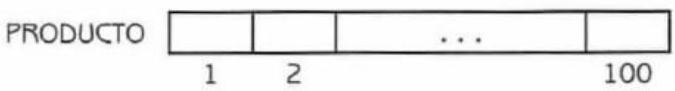
Simule por medio de un diagrama de flujo un directorio telefónico que lea nombres y números telefónicos en dos arreglos paralelos. Los nombres están ordenados alfabéticamente en orden ascendente.

Cuando el usuario teclee un nombre, el programa debe verificar si éste ya estaba en el directorio, en cuyo caso imprimirá el número telefónico. De no ser así y si hay capacidad, pedirá al usuario que escriba el teléfono y agregará nombre y número a los arreglos.

Datos: NOMBRE [1..N], TELEFONO [1..N] (arreglos unidimensionales de tipo cadena de caracteres, $1 \leq N \leq 1000$).

PS 4.36

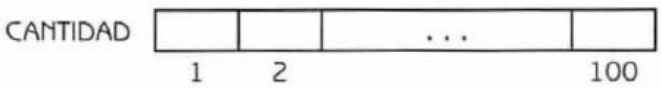
Una tienda vende como máximo 100 productos diferentes. La información sobre los productos se encuentra almacenada en los siguientes arreglos:



Donde:

PRODUCTO[i] representa la clave del producto i.

PRODUCTO[1..N], $1 \leq N \leq 100$, es un arreglo unidimensional ordenado de tipo entero que almacena las claves de los productos.



Donde:

CANTIDAD[i] representa la cantidad existente del producto i.

www.FreeLibros.com

CANTIDAD[1..N], con $1 \leq N \leq 100$, es un arreglo unidimensional de tipo entero que almacena la cantidad existente de los productos.

Construya un diagrama de flujo que haga lo siguiente:

a) Lea el número actual N de productos en existencia.
b) Lea los dos arreglos.
c) Actualice los arreglos de acuerdo a las siguientes transacciones:

$CLAOP_1$, $CLAPRO_1$, $CANT_1$
$CLAOP_2$, $CLAPRO_2$, $CANT_2$
. . .
X, 0, 0

Donde:

$CLAOP_i$ es una variable de tipo caracter que expresa el tipo de operación i. El tipo de operación puede ser:

"C" que expresa compra, en cuyo caso deberá incrementar la existencia del producto si el mismo existe. En caso contrario deberá incorporarlo.
"V", en cuyo caso deberá verificar si existe la cantidad necesaria para satisfacer el pedido. Si es posible satisfacerlo se decrementará la cantidad en existencia, en caso contrario se debe emitir un mensaje adecuado.

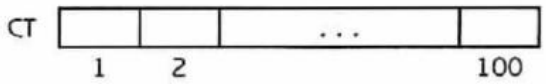
$CLAPRO_j$ es una variable de tipo entero que expresa el tipo de producto j involucrado en la operación i.

$CANT_j$ es una variable de tipo entero que representa la cantidad del producto j, involucrado en la operación i.

El fin de datos de las transacciones está dado por: X, 0, 0.

PS 4.37

Se tienen dos arreglos que contienen la siguiente información:

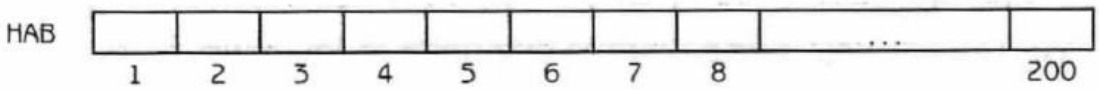


Donde:

CT[i] representa el nombre del centro turístico i.

www.FreeLibros.com

Nota: El arreglo CT está ordenado de manera creciente.



Donde:

HAB[i] y HAB[i + 1] representan el total de habitaciones simples y dobles, respectivamente, del centro turístico i. Es decir, a cada elemento de CT le corresponden dos elementos de HAB.

Construya un diagrama de flujo que luego de leer los arreglos, pueda llevar a cabo las siguientes operaciones según las claves que se dan como datos:

CLAVE	*OPERACION*
L	Lista el nombre de los centros turísticos del país.
R	Dado el nombre de un centro turístico como dato, informa el total, por categoría, de habitaciones de dicho centro turístico.
I	Se crea un nuevo centro turístico en el país. Se deben actualizar los arreglos que correspondan.
M	Dado el nombre de un centro turístico, determinar cuál es el que tiene mayor número de habitaciones (contabilizando las simples y las dobles).

Datos: CT[1..N], HAB[1..N * 2], CLAVE
$1 \leq N \leq 100$

Donde:

CT es un arreglo unidimensional de tipo cadena de caracteres. Almacena el nombre de los centros turísticos.

HAB es un arreglo unidimensional de tipo entero. Almacena información sobre los tipos de habitaciones de cada centro.

CLAVE es una variable de tipo caracter que representa el tipo de operación que se desea realizar.

www.FreeLibros.com

PS 5.1

Una compañía distribuye N productos a distintos comercios de la ciudad. Para ello almacena en un arreglo toda la información relacionada a su mercancía:

- Clave.
- Descripción.
- Existencia.
- Mínimo a mantener en existencia.
- Precio unitario.

Escriba un diagrama de flujo que pueda llevar a cabo las siguientes operaciones:

a) Venta de un producto: Se deben actualizar los campos que correspondan y verificar que la nueva existencia no esté por debajo del mínimo. (Datos: clave, cantidad vendida).
b) Reabastecimiento de un producto: Se deben actualizar los campos que correspondan (Datos: clave, cantidad comprada).
c) Actualizar el precio de un producto (Datos: clave, porcentaje de aumento).
d) Informar sobre un producto: Se deben proporcionar todos los datos relacionados a un producto (Dato: clave).

Dato: PRODUCTOS[1..N] $1 \leq N \leq 150$

Donde:

PRODUCTOS: es un arreglo unidimensional de registros. Cada registro tiene los siguientes campos:
CLA: Variable de tipo entero. Representa la clave del producto.
DES: Variable de tipo cadena de caracteres. Expresa la descripción del producto.
EXI: Variable de tipo entero. Representa la existencia del producto.
MIN: Variable de tipo entero. Representa el mínimo a mantener en existencia del producto con clave CLA.
PU: Variable de tipo real. Expresa el precio unitario del producto.

PS 5.2

En una escuela se almacena información sobre los alumnos en un arreglo unidimensional de registros denominado ESC. Por cada alumno se almacena la siguiente información:

www.FreeLibros.com

- Nombre del alumno: Cadena de caracteres.
- Matrícula del alumno: Entero.
- Número de semestres cursados: Entero.
- Calificación promedio por semestre: Arreglo unidimensional de tipo real. Su capacidad máxima es para 9 elementos.
- Carrera: Cadena de caracteres.

Construya un diagrama de flujo tal que dada la información de N alumnos, pueda efectuar las siguientes operaciones:

a) Listar nombre y matrícula de los estudiantes que tengan un promedio general mayor a 9.
b) Listar nombre y matrícula de los estudiantes, de la carrera de Economía, que hayan obtenido un promedio superior a 9 en *todos* los semestres cursados hasta el momento.
c) Escribir el nombre y matrícula del alumno que tiene el mayor promedio de la carrera Ingeniería en Computación.

Dato: ESC[1...N] $1 \leq N \leq 2000$

Donde:

ESC: es un arreglo unidimensional de registros. Cada registro tiene los siguientes campos:
NOM: Variable de tipo cadena de caracteres que representa el nombre del alumno.
MAT: Variable de tipo entero que representa la matrícula del alumno.
NSE: Variable de tipo entero que representa el número de semestres cursados por el alumno.
SEM: Arreglo unidimensional de tipo real. Almacena el promedio de calificaciones en cada semestre del alumno.
CAR: Variable de tipo cadena de caracteres que representa la carrera en la que está inscrito el alumno.

PS 5.3

Una inmobiliaria tiene información disponible sobre departamentos en renta. De cada departamento se conoce:

CLAVE: Variable entera que identifica el inmueble.
EXTENSION: Variable de tipo real que representa la superficie del departamento, en metros cuadrados.

www.FreeLibros.com

UBICACION: Variable de tipo cadena de caracteres que indica si la ubicación del inmueble es excelente, buena, regular o mala.
PRECIO: Variable de tipo real que representa el precio del inmueble.
DISPONIBLE: Variable de tipo booleana que indica si el inmueble está disponible para la renta o no.

Diariamente acuden muchos clientes a la inmobiliaria solicitando información. Escriba un diagrama de flujo capaz de realizar las siguientes operaciones:

a) Listar los datos de todos los departamentos disponibles que tengan un precio inferior o igual a un cierto valor P.
b) Listar los datos de los departamentos disponibles que tengan una superficie mayor o igual a un cierto valor dado E y una ubicación excelente.
c) Listar el monto de la renta de todos los departamentos alquilados.

PS 5.4

En un negocio mayorista se tiene almacenada en un arreglo de registros información de los clientes. La estructura de este arreglo es la siguiente :

CLIENTE

CLAVE	NOMBRE	SALDO	CLAVE	NOMBRE	SALDO	...	CLAVE	NOMBRE	SALDO
	1			2				300	

Donde:

CLIENTE[i] Representa al registro que contiene información del cliente i. Los campos del registro son los siguientes:
CLAVE: Representa la clave del cliente. Es una variable de tipo entero.
NOMBRE: Variable de tipo cadena de caracteres que representa el nombre del cliente.
SALDO: Variable de tipo real que representa el saldo.

Nota: El arreglo está ordenado teniendo en cuenta la clave del cliente.

Al finalizar el día se debe actualizar esta información, con base en las transacciones realizadas durante el día. Las mismas se expresan de esta forma:

N
CLA_1, $TIPO_1$, $MONTO_1$

www.FreeLibros.com

CLA_2, $TIPO_2$, $MONTO_2$

...

CLA_N, $TIPO_N$, $MONTO_N$

Donde:

N representa el número de transacciones realizadas durante el día. Variable de tipo entero.

CLA_i representa la clave del cliente de la transacción$_i$. Es una variable de tipo entero.

$TIPO_i$ representa el tipo de operación efectuada en la transacción$_i$. El tipo puede ser una alta (A), baja (B) o compra (C). Variable de tipo caracter.

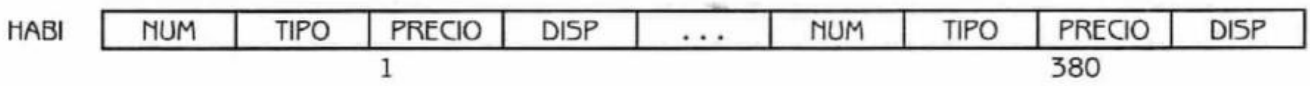
$MONTO_i$ expresa el monto de la transacción i. Si es una baja, el monto aparece en 0. Si es un pago, el monto se expresa negativamente. Por ejemplo si el cliente con clave 1250 pagó $680 a su cuenta, la transacción aparece como : 1250 P -680 . Variable de tipo real.

Construya un diagrama de flujo que pueda efectuar las siguientes operaciones:

a) Actualice el arreglo de registros con base en las transacciones realizadas durante el día.
b) Escriba el nombre de los clientes cuyo saldo es mayor a $10 000.
c) Escriba el arreglo actualizado.

PS 5.5

En un hotel se tiene almacenada información sobre las habitaciones y huéspedes, en los siguientes arreglos de registros.



Donde:

HABI[i] representa al registro que contiene información de la habitación i. Los campos del registro son los siguientes:

NUM: representa el número de habitación. Variable de tipo entero.

TIPO: indica el tipo de habitación. Variable de tipo cadena de caracteres se ingresa:

SI: Si la habitación es simple.
DO: Si la habitación es doble.
TR: Si la habitación es triple.
SU: Si la habitación es suite.

www.FreeLibros.com

PRECIO: Representa el precio de la habitación. Variable de tipo real.
DISP: Indica si la habitación está disponible o no. Variable de tipo cadena de caracteres. Se ingresa:

SI: Si la habitación está disponible.
NO: Si la habitación no está disponible.

Nota: El arreglo está ordenado teniendo en cuenta el número de la habitación.

HUES

NUMHAB	NOM	FECARR	NUMHAB	NOM	FECARR	. . .	
1				2			380

Donde:

HUES[i] representa al registro que contiene la información del huésped i. Los campos del registro son los siguientes:
NUMHAB: Indica el número de habitación. Variable de tipo entero.
NOM: Expresa el nombre del huésped. Variable de tipo cadena de caracteres.
FECARR: Indica la fecha del arribo del cliente. Variable de tipo real.

Construya un diagrama de flujo tal que luego de leer los dos arreglos, pueda efectuar las siguientes operaciones:

a) Dado el nombre de un huésped y la fecha actual, regrese lo que debe pagar dicho huésped (Asuma que la diferencia entre las fechas la da el número de días de alojamiento).
b) Obtenga el total de habitaciones simples, dobles, triples y suites que hay disponibles.
c) Dado el nombre de un huésped, lo elimine de los arreglos correspondientes.
d) Dado el nombre de un huésped que desea una habitación tipo X, se dé de alta en los arreglos correspondientes si existe habitación disponible. Deberá además leer la fecha del día de hoy.

PS 5.6

El departamento de personal de una escuela tiene registros del nombre, sexo y edad de cada uno de los profesores adscritos al mismo.

NOMBRE	EDAD	SEXO	NOMBRE	EDAD	SEXO	. . .	NOMBRE	EDAD	SEXO
1				2				N	

www.FreeLibros.com

Escriba un programa que calcule e imprima lo siguiente:

a) Edad promedio del grupo de profesores.
b) Nombre del profesor más joven del grupo.
c) Nombre del profesor de más edad.
d) Número de profesoras con edad mayor al promedio.
e) Número de profesores con edad menor al promedio.

Datos: PERSONAL[1..N] (arreglo unidimensional de registros. NOMBRE es un campo del registro de tipo cadena de caracteres. EDAD es un campo del registro de tipo entero. SEXO es otro campo del registro de tipo caracter. Se ingresa "F" si es mujer y "M" si es hombre, $1 \leq N \leq 100$).

www.FreeLibros.com

Indice

A

Acceso a los campos de un registro, 346
Algoritmos, diagramas de flujo y programas, 1
  conceptos fundamentales, 8
    bloque de asignación, 22
    expresiones lógicas, 15
    identificadores, constantes y variables, 9
    operaciones aritméticas, 12
    tipos de datos, 8
  diagramas de flujo, 4
  problemas y algoritmos, 1
Arreglos, 175
  bidimensionales, 210, 446
  declaración de arreglos bidimensionales, 212
  de más de dos dimensiones, 222, 453
  de registros, 347
  desordenados, 190
  multidimensionales, 209
  ordenados, 198
  paralelos, 454
  unidimensionales, 181, 441

C

Combinaciones entre arreglos y registros, 347
Conceptos fundamentales, 8
Construcción de diagramas de flujo, 24

D

Definición de registros, 344
Diagramas de flujo, 4
  conceptos fundamentales, 8
    tipos de datos, 8
  reglas para la construcción de, 4

E

Estructuras algorítmicas repetitivas, 107
  mientras (WHILE), 115
  repetir (FOR), 107
Estructuras algorítmicas selectivas, 53
  estructura selectiva doble *si entonces/sino*, 57
  estructura selectiva múltiple *si múltiple*, 62
  estructura selectiva simple *si entonces*, 54
  estructuras selectivas en cascada (anidadas), 70
  introducción, 53
Estructuras de datos, 175, 343
  arreglos, 175
    bidimensionales, 210
    definición de arreglos, 182
    multidimensionales, 209
    operaciones con arreglos, 184
  registros, 343
    acceso a los campos de un, 346
    combinaciones entre arreglos y registros, 347
    definición de, 344
    diferencias con arreglos, 347

J

Jerarquía de los operadores aritméticos, 12

O

Operación con arreglos bidimensionales, 215
Operaciones con arreglos, 184
  actualización, 190
  asignación, 188
  escritura, 187

www.FreeLibros.com

lectura, 185
Operadores
lógicos, 18
relacionales, 16

P

Programas, 31

R

Registros
anidados, 350
con arreglos, 352

S

Símbolos utilizados en los diagramas de flujo, 5

www.FreeLibros.com

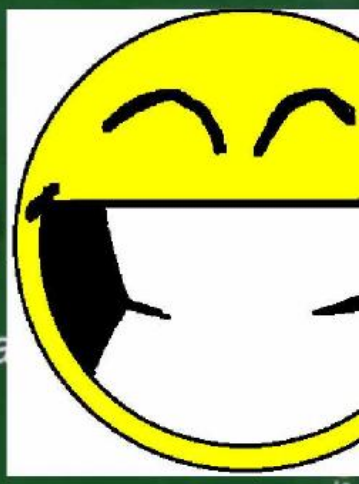
Esta edición se terminó de imprimir en agosto de 2006. Publicada por ALFAOMEGA GRUPO EDITOR, S.A. de C.V. Apartado Postal 73-267, 03311, México, D.F. La impresión se realizó en REPROFLO, S.A. de C.V. Chipiona 115, Col. Cerro de la Estrella, Iztapalapa, 09880, México, D.F.

www.FreeLibros.com

Metodología de programación

Algoritmos, diagramas de flujo y programa

3ª Edición



¿Podemos enseñar a analizar un problema?
¿Podemos enseñar a razonar flexiblemente?
¿Podemos enseñar a pensar?

Las respuestas innegablemente no son fáciles de obtener. Sin embargo, el esfuerzo y la búsqueda constante del saber siempre valen la pena. Encontrar caminos es una tarea estimulante. Enseñar al alumno a ser flexible, a observar un problema desde ángulos diferentes, a ver distintas alternativas, a *sentir* la belleza de una solución es una tarea de maestros con pasión. **Enseñar a aprender. Enseñar a ver. Enseñar a saber.**

Esta edición reúne lo mejor de las ediciones anteriormente publicadas con la misma editorial, e incluye nuevos problemas, mejores soluciones y novedosos componentes que permiten una mejor comprensión, claridad y aplicación de los conceptos estudiados. El lenguaje que se utiliza para mostrar la solución de los problemas es *pseudo-código de alto nivel*. Esta característica es muy importante porque facilita al alumno concentrarse en las estructuras de datos y algoritmos asociados a ellas sin la necesidad de relacionarlos a un lenguaje de programación en particular.

Alfaomega Grupo Editor

www.FreeLibros.com